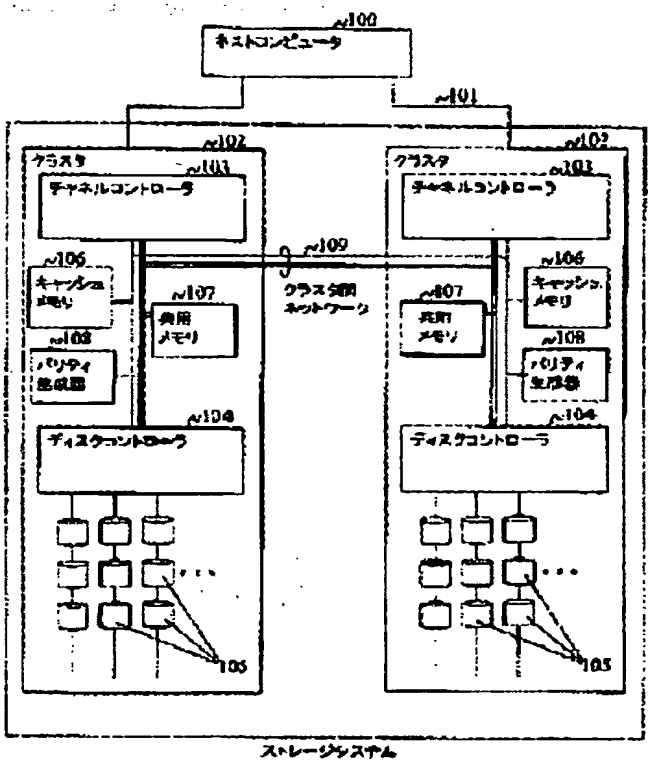
CONFIGURATION OF RAID AMONG CLUSTERS IN CLUSTER CONFIGURING STORAGE

Patent number:	JP2003131818
Publication date:	2003-05-09
Inventor:	YAMAMOTO YASUTOMO; OEDA TAKASHI; SATO TAKAO
Applicant:	HITACHI LTD
Classification:	
- international:	G06F3/06; G06F12/08
- european:	
Application number:	JP20010327103 20011025
Priority number(s):	JP20010327103 20011025

Abstract of **JP2003131818**

PROBLEM TO BE SOLVED: To realize load distribution among clusters in a cluster configuration storage utilizing its characteristics. SOLUTION: A RAID stretching over a plurality of clusters is constructed using a plurality of disk units under a plurality of clusters. In creating and writing redundant data such as mirror data and parity, an update value or a value before update of data held in the cache memory of each cluster is directly used through the network connecting each cluster. This method prevents efficiency of the cache memory from being deteriorated since reproduction of update data is no conducted on the cache memory.

図 1



(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2003－131818
（P2003－131818A）
(43)公開日　平成15年5月9日(2003.5.9)

(51)Int.Cl.'		識別記号	FI		テーマコード'(参考)
Ｇ０６Ｆ	3/06	３０２	Ｃ０６Ｆ	3/06	３０２Ａ　５Ｂ００５
		３０１			３０１Ｓ　５Ｂ０６５
		３０５			３０５Ｃ
		５４０			５４０
	12/08	５０１		12/08	５０１Ｅ

審査請求　未請求　請求項の数８　ＯＬ　（全 17 頁）　最終頁に続く

(21)出願番号　特願2001－327103(P2001－327103)

(22)出願日　平成13年10月25日(2001.10.25)

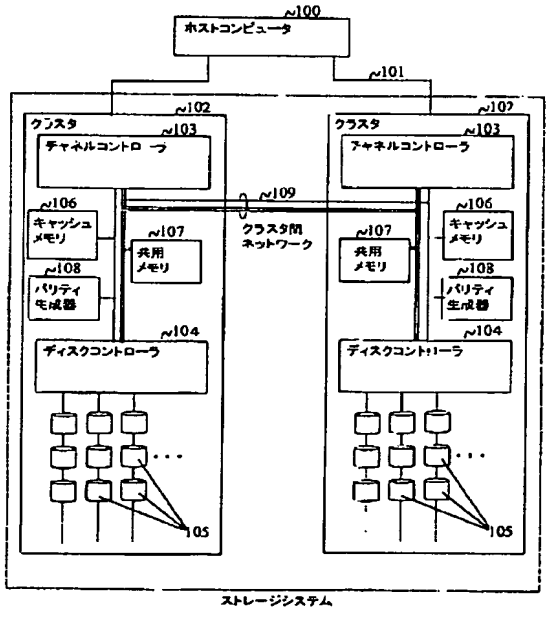
(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地
(72)発明者　山本　康友
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(72)発明者　大枝　高
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(74)代理人　100075096
弁理士　作田　康夫

最終頁に続く

(54)【発明の名称】　クラスタ構成ストレージにおけるクラスタ間ＲＡＩＤ構成

(57)【要約】
【課題】クラスタ構成ストレージにおいて、その特性を生かしたクラスタ間負荷分散を実現する。
【解決手段】複数のクラスタ配下のディスク装置複数台を用いて、複数クラスタにまたがるＲＡＩＤを構成する。ミラーデータやパリティなど冗長データの作成および書き出し時には、クラスタ間を接続するネットワークを介して、各クラスタのキャッシュメモリに保持されたデータの更新値や更新前の値を直接用いる。キャッシュメモリ上で更新データの複製を作成しないため、キャッシュメモリの利用効率の低下を回避できる。

図 1



【特許請求の範囲】
【請求項１】　１台以上のホストコンピュータと複数のクラスタからなるストレージシステムを接続してなる計算機システムであって、
ストレージクラスタが１台以上の記憶装置と１つ以上のコントローラとキャッシュメモリと制御情報を格納する共用メモリを有し、クラスタ間を接続するネットワークにより各コントローラが他のクラスタ内のキャッシュメモリや共用メモリの内容を利用可能である計算機システムにおいて、
複数のクラスタの１台以上の記憶装置からなる記憶装置アレイを構成し、
当該記憶装置アレイに対するデータ更新時には、第１のクラスタのキャッシュメモリに保持した更新データを用いて第２のクラスタの記憶装置に格納された冗長データの更新を行うことを特徴とするストレージシステム。
【請求項２】　請求項１記載の計算機システムにおいて、
第１のクラスタが有する１台以上の第１の記憶装置のデータの複製を、第２のクラスタが有する１台以上の第２の記憶装置へ格納し、
当該記憶装置アレイへのデータ更新時には、第１のクラスタのキャッシュメモリに保持した更新データを第１の記憶装置と第２の記憶装置へ書き込むことを特徴とするストレージシステム。
【請求項３】　請求項１記載の計算機システムにおいて、
複数のクラスタが有する１台以上の記憶装置を集めた記憶装置（ｎ＋１）台に、一定単位のデータ毎にｎ個のデータと、対応するパリティを格納し、
当該記憶装置アレイへのデータ更新時には、第１のクラスタのキャッシュメモリに保持した更新データの更新値と、第１の記憶装置から第１のクラスタのキャッシュメモリに読み上げた当該データの更新前の値と、第２の記憶装置から第２のクラスタのキャッシュメモリに読み上げたパリティの更新前の値を用いてパリティの更新値を生成し、
当該データの更新値とパリティの更新値をそれぞれ第１の記憶装置と第２の記憶装置に書き込むことを特徴とするストレージシステム。
【請求項４】　請求項３記載のストレージシステムにおいて、
パリティ生成時に必要なデータを保持している量に従って、パリティ生成を行うクラスタを決定することを特徴とするストレージシステム。
【請求項５】　請求項１記載の計算機システムにおいて、
１クラスタ内の１台以上の記憶装置からなる第１の記憶装置アレイと、
複数のクラスタの１台以上の記憶装置からなる第２の記憶装置アレイが存在し、
第１と第２の記憶装置アレイへのホストコンピュータからのアクセスを受けたままの状態で、両アレイ間でデータを入れかえることを特徴とするストレージシステム。
【請求項６】　請求項１記載の計算機システムにおいて、
１クラスタ内の１台以上の記憶装置からなる第１の記憶装置アレイに対して、
複数のクラスタの１台以上の記憶装置からなるホスト未使用の第２の記憶装置アレイを作成し、
第１の記憶装置アレイへのホストコンピュータからのアクセスを受けたままの状態で、第１の記憶装置アレイのデータを第２の記憶装置アレイへ移動することを特徴とするストレージシステム。
【請求項７】　請求項１記載の計算機システムにおいて、
複数のクラスタの１台以上の記憶装置からなるホスト未使用の第１の記憶装置アレイを作成し、
１クラスタ内の１台以上の記憶装置からなる第２の記憶装置アレイに対して、
第１の記憶装置アレイへのホストコンピュータからのアクセスを受けたままの状態で、第１の記憶装置アレイのデータを第２の記憶装置アレイへ移動することを特徴とするストレージシステム。
【請求項８】　１台以上のホストコンピュータと複数のストレージが接続してなる計算機システムであって、
ストレージが１台以上のディスク装置と１つ以上のコントローラとキャッシュメモリと制御情報を格納する共用メモリを有し、ストレージ間を接続するネットワークにより各コントローラが他のストレージ内のキャッシュメモリや共用メモリの内容を利用可能である計算機システムにおいて、
複数のストレージの１台以上のディスク装置からなるＲＡＩＤを構成することを特徴とするストレージシステム。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、ホストコンピュータと記憶装置を接続してなる計算機システム、特に複数クラスタで構成されるストレージ間での負荷分散方法に関する。
【０００２】
【従来の技術】近年、計算機で取り扱われるデータ量は飛躍的に増大し、それに従ってストレージの大容量化が進んでいる。ストレージの大容量化は機器導入コストおよび管理コストの増大をまねき、各ストレージベンダはコスト低減が必須命題となっている。大容量ストレージのコスト低減を実現する方法の一つとして、近年提唱されているＳＡＮ(Ｓｔｏｒａｇｅ Ａｒｅａ Ｎｅｔｗｏｒｋ)やＮＡＳ(Ｎｅｔｗｏｒｋ Ａｔｔａｃｈｅｄ Ｓ

【特許請求の範囲】
【請求項１】　１台以上のホストコンピュータと複数のクラスタからなるストレージシステムを接続してなる計算機システムであって、
ストレージクラスタが１台以上の記憶装置と１つ以上のコントローラとキャッシュメモリと制御情報を格納する共用メモリを有し、クラスタ間を接続するネットワークにより各コントローラが他のクラスタ内のキャッシュメモリや共用メモリの内容を利用可能である計算機システムにおいて、
複数のクラスタの１台以上の記憶装置からなる記憶装置アレイを構成し、
当該記憶装置アレイに対するデータ更新時には、第１のクラスタのキャッシュメモリに保持した更新データを用いて第２のクラスタの記憶装置に格納された冗長データの更新を行うことを特徴とするストレージシステム。
【請求項２】　請求項１記載の計算機システムにおいて、
第１のクラスタが有する１台以上の第１の記憶装置のデータの複製を、第２のクラスタが有する１台以上の第２の記憶装置へ格納し、
当該記憶装置アレイへのデータ更新時には、第１のクラスタのキャッシュメモリに保持した更新データを第１の記憶装置と第２の記憶装置へ書き込むことを特徴とするストレージシステム。
【請求項３】　請求項１記載の計算機システムにおいて、
複数のクラスタが有する１台以上の記憶装置を集めた記憶装置（ｎ＋１）台に、一定単位のデータ毎にｎ個のデータと、対応するパリティを格納し、
当該記憶装置アレイへのデータ更新時には、第１のクラスタのキャッシュメモリに保持した更新データの更新値と、第１の記憶装置から第１のクラスタのキャッシュメモリに読み上げた当該データの更新前の値と、第２の記憶装置から第２のクラスタのキャッシュメモリに読み上げたパリティの更新前の値を用いてパリティの更新値を生成し、
当該データの更新値とパリティの更新値をそれぞれ第１の記憶装置と第２の記憶装置に書き込むことを特徴とするストレージシステム。
【請求項４】　請求項３記載のストレージシステムにおいて、
パリティ生成時に必要なデータを保持している量に従って、パリティ生成を行うクラスタを決定することを特徴とするストレージシステム。
【請求項５】　請求項１記載の計算機システムにおいて、
１クラスタ内の１台以上の記憶装置からなる第１の記憶装置アレイと、
複数のクラスタの１台以上の記憶装置からなる第２の記憶装置アレイが存在し、
第１と第２の記憶装置アレイへのホストコンピュータからのアクセスを受けたままの状態で、両アレイ間でデータを入れかえることを特徴とするストレージシステム。
【請求項６】　請求項１記載の計算機システムにおいて、
１クラスタ内の１台以上の記憶装置からなる第１の記憶装置アレイに対して、
複数のクラスタの１台以上の記憶装置からなるホスト未使用の第２の記憶装置アレイを作成し、
第１の記憶装置アレイへのホストコンピュータからのアクセスを受けたままの状態で、第１の記憶装置アレイのデータを第２の記憶装置アレイへ移動することを特徴とするストレージシステム。
【請求項７】　請求項１記載の計算機システムにおいて、
複数のクラスタの１台以上の記憶装置からなるホスト未使用の第１の記憶装置アレイを作成し、
１クラスタ内の１台以上の記憶装置からなる第２の記憶装置アレイに対して、
第１の記憶装置アレイへのホストコンピュータからのアクセスを受けたままの状態で、第１の記憶装置アレイのデータを第２の記憶装置アレイへ移動することを特徴とするストレージシステム。
【請求項８】　１台以上のホストコンピュータと複数のストレージが接続してなる計算機システムであって、
ストレージが１台以上のディスク装置と１つ以上のコントローラとキャッシュメモリと制御情報を格納する共用メモリを有し、ストレージ間を接続するネットワークにより各コントローラが他のストレージ内のキャッシュメモリや共用メモリの内容を利用可能である計算機システムにおいて、
複数のストレージの１台以上のディスク装置からなるＲＡＩＤを構成することを特徴とするストレージシステム。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】本発明は、ホストコンピュータと記憶装置を接続してなる計算機システム、特に複数クラスタで構成されるストレージ間での負荷分散方法に関する。
【０００２】
【従来の技術】近年、計算機で取り扱われるデータ量は飛躍的に増大し、それに従ってストレージの大容量化が進んでいる。ストレージの大容量化は機器導入コストおよび管理コストの増大をまねき、各ストレージベンダはコスト低減が必須命題となっている。大容量ストレージのコスト低減を実現する方法の一つとして、近年提唱されているＳＡＮ(Ｓｔｏｒａｇｅ Ａｒｅａ Ｎｅｔｗｏｒｋ)やＮＡＳ(Ｎｅｔｗｏｒｋ Ａｔｔａｃｈｅｄ Ｓ

torage）といった、複数のストレージ機器を組合せて大規模なストレージシステムを構築する方法がある。異なるベンダのストレージ機器を接続、統合管理できるなど、導入コストおよび管理コスト低減に効果的なソリューションである。ただ、異なるベンダのストレージ機器を扱う都合上、互換性検証や統合管理のための基準策定など実現にはいくつかの課題がある。

【０００３】これとは別の実現方法として、従来汎用機用として用いられてきた大型ストレージを採用する方法も考えられる。大型ストレージは、汎用機用ストレージとしての実績より培った高性能性、高信頼性、高可用性を基本要素とし、市場の大容量化に合わせて、記憶容量の増大、接続ホストインタフェースの拡張を行ってきた。ユーザが必要とする大規模な記憶容量を一台の機器で提供できるため、複数台のストレージ機器でシステム構築する場合と比較し、より少ない設置面積で、かつストレージの管理コストを低減可能である。また、ＳＡＮやＮＡＳと比べて、高信頼なストレージシステムとしての長い実績が大きな長所となる。

【０００４】ただ、一台のストレージ機器で大容量かつ多ホスト接続性を実現させようとする場合、内部に実装するプロセッサや記憶装置数の増大が必要となり、それら機器間での高速連携のため、内部バスや共用メモリなどの高速化や、プロセッサ間での競合回避などが、ハードウェアおよびソフトウェア上での技術面およびコスト面での大きな課題となる。この課題を解決し、大規模かつ低コストな大型ストレージを構築する方法として、クラスタ技術の適用が考えられる。クラスタ技術はこれまで主にサーバなどホストコンピュータの分野で、大量の処理能力を実現する実装方式として用いられてきたが、これをストレージに適用することで、大規模なストレージを比較的低コストで実装することが可能となる。

【０００５】クラスタ構成大型ストレージでは、各クラスタ毎にホストインタフェース、コントローラ、記憶装置、キャッシュメモリなどを搭載し、各クラスタが独立したストレージとして動作することが可能となる。クラスタ構成ストレージが一台のストレージとして動作するためには、クラスタ間を相互接続するネットワークが別途必要となる。各クラスタの持つホストインタフェースの上位にスイッチを実装し、各クラスタの記憶装置へのアクセスを振り分けてもよいし、クラスタ間を相互接続し、互いのキャッシュメモリなどを相互アクセス可能なネットワークを実装してもよい。ただ、ストレージアクセスの傾向にも依存するが、比較的ヒット率の高いアクセスなどの場合、記憶媒体自体よりもアクセスを制御するコントローラの処理能力自体が性能上ボトルネックとなることが多い。このようなケースでの性能向上のためにシステムに実装された資源を有効利用するには、クラスタ間コントローラでの処理分散を実現することが望ましい。そのためには、後者の、クラスタ間相互接続ネットワークを具備することが望ましいと考えられる。また、クラスタ間データコピーなど各種機能をサポートするにも、同ネットワークの具備は必須である。よって、以降では、クラスタ構成ストレージにはクラスタ間での相互アクセスを可能とするネットワークが実装されていることを前提とする。

【０００６】クラスタ構成ストレージでは複数クラスタ間での負荷分散を実現可能である。

【０００７】ストレージの負荷分散については、ＵＳＰ５８３２２２２において、異なるストレージに搭載された複数のディスク装置間でＲＡＩＤ(Ｒｅｄｕｎｄａｎｔ Ａｒｒａｙ ｏｆ Ｉｎｄｅｐｅｎｄｅｎｔ Ｄｉｓｋ)を構成する技術が開示されている。複数ストレージにまたがるＲＡＩＤへデータを格納することで、データアクセス時の起動ディスク装置数および動作するコントローラ数を増加し、負荷分散を図ることが可能となる。

【０００８】

【発明が解決しようとする課題】クラスタ構成ストレージの負荷分散のため、ＵＳＰ５８３２２２２で開示されている技術を単純に適用しても、クラスタ構成の特性を有効に利用できないと考えられる。理由は次の通りである。

【０００９】ＵＳＰ５８３２２２２で開示されている技術では、独立した複数台のストレージ間でＲＡＩＤを構成しているため、ストレージ間でのデータの授受では、ストレージ間を接続するネットワークを介して、送信側のキャッシュメモリへデータを書き込む必要がある。例えば、２台のストレージ配下のディスク装置各１台、計２台でＲＡＩＤレベル１のＲＡＩＤを構築した場合、当該ディスク装置へのデータ更新を受けた一方のストレージ甲から、他方のストレージ乙へ当該更新データを転送する必要がある。このとき、ストレージ甲から転送されたデータはストレージ乙のキャッシュメモリへ格納されるため、システム全体で見ると、一時的にキャッシュ上でデータが二重に保持され、キャッシュメモリの使用効率が低下する。特に、ストレージ乙側で受信した更新データを一定期間保持した後、ディスク装置に反映するような制御を行う場合、キャッシュメモリ使用効率の低下が著しいと予想される。

【００１０】また、ＵＳＰ５８３２２２で開示されている技術で、ＲＡＩＤレベル５のＲＡＩＤを構成する場合、更新データに対してパリティ(冗長データ)を生成するために必要なデータを、当該データに対応するパリティを格納しているストレージへ転送する必要がある。ＲＡＩＤレベル５のＲＡＩＤでは、一定の単位(ストライプ)毎にデータを分割し、複数のディスク装置に格納し(ストライピング)、一列のデータストライプに対して、１つ以上のパリティを生成し、データとは別のディスク装置へ格納する。パリティを格納するディスク装置はストライプ列毎に変えることで、データ更新時のパリティ

torage）といった、複数のストレージ機器を組合せて大規模なストレージシステムを構築する方法がある。異なるベンダのストレージ機器を接続、統合管理できるなど、導入コストおよび管理コスト低減に効果的なソリューションである。ただ、異なるベンダのストレージ機器を扱う都合上、互換性検証や統合管理のための基準策定など実現にはいくつかの課題がある。

【０００３】これとは別の実現方法として、従来汎用機用として用いられてきた大型ストレージを採用する方法も考えられる。大型ストレージは、汎用機用ストレージとしての実績より培った高性能性、高信頼性、高可用性を基本要素とし、市場の大容量化に合わせて、記憶容量の増大、接続ホストインタフェースの拡張を行ってきた。ユーザが必要とする大規模な記憶容量を一台の機器で提供できるため、複数台のストレージ機器でシステム構築する場合と比較し、より少ない設置面積で、かつストレージの管理コストを低減可能である。また、ＳＡＮやＮＡＳと比べて、高信頼なストレージシステムとしての長い実績が大きな長所となる。

【０００４】ただ、一台のストレージ機器で大容量かつ多ホスト接続性を実現させようとする場合、内部に実装するプロセッサや記憶装置数の増大が必要となり、それら機器間での高速連携のため、内部バスや共用メモリなどの高速化や、プロセッサ間での競合回避などが、ハードウェアおよびソフトウェア上での技術面およびコスト面での大きな課題となる。この課題を解決し、大規模かつ低コストな大型ストレージを構築する方法として、クラスタ技術の適用が考えられる。クラスタ技術はこれまで主にサーバなどホストコンピュータの分野で、大量の処理能力を実現する実装方式として用いられてきたが、これをストレージに適用することで、大規模なストレージを比較的低コストで実装することが可能となる。

【０００５】クラスタ構成大型ストレージでは、各クラスタ毎にホストインタフェース、コントローラ、記憶装置、キャッシュメモリなどを搭載し、各クラスタが独立したストレージとして動作することが可能となる。クラスタ構成ストレージが一台のストレージとして動作するためには、クラスタ間を相互接続するネットワークが別途必要となる。各クラスタの持つホストインタフェースの上位にスイッチを実装し、各クラスタの記憶装置へのアクセスを振り分けてもよいし、クラスタ間を相互接続し、互いのキャッシュメモリなどを相互アクセス可能なネットワークを実装してもよい。ただ、ストレージアクセスの傾向にも依存するが、比較的ヒット率の高いアクセスなどの場合、記憶媒体自体よりもアクセスを制御するコントローラの処理能力自体が性能上ボトルネックとなることが多い。このようなケースでの性能向上のためにシステムに実装された資源を有効利用するには、クラスタ間コントローラでの処理分散を実現することが望ましい。そのためには、後者の、クラスタ間相互接続ネットワークを具備することが望ましいと考えられる。また、クラスタ間データコピーなど各種機能をサポートするにも、同ネットワークの具備は必須である。よって、以降では、クラスタ構成ストレージにはクラスタ間での相互アクセスを可能とするネットワークが実装されていることを前提とする。

【０００６】クラスタ構成ストレージでは複数クラスタ間での負荷分散を実現可能である。

【０００７】ストレージの負荷分散については、ＵＳＰ５８３２２２２において、異なるストレージに搭載された複数のディスク装置間でＲＡＩＤ(Ｒｅｄｕｎｄａｎｔ Ａｒｒａｙ ｏｆ Ｉｎｄｅｐｅｎｄｅｎｔ Ｄｉｓｋ)を構成する技術が開示されている。複数ストレージにまたがるＲＡＩＤへデータを格納することで、データアクセス時の起動ディスク装置数および動作するコントローラ数を増加し、負荷分散を図ることが可能となる。

【０００８】

【発明が解決しようとする課題】クラスタ構成ストレージの負荷分散のため、ＵＳＰ５８３２２２２で開示されている技術を単純に適用しても、クラスタ構成の特性を有効に利用できないと考えられる。理由は次の通りである。

【０００９】ＵＳＰ５８３２２２２で開示されている技術では、独立した複数台のストレージ間でＲＡＩＤを構成しているため、ストレージ間でのデータの授受では、ストレージ間を接続するネットワークを介して、送信側のキャッシュメモリへデータを書き込む必要がある。例えば、２台のストレージ配下のディスク装置各１台、計２台でＲＡＩＤレベル１のＲＡＩＤを構築した場合、当該ディスク装置へのデータ更新を受けた一方のストレージ甲から、他方のストレージ乙へ当該更新データを転送する必要がある。このとき、ストレージ甲から転送されたデータはストレージ乙のキャッシュメモリへ格納されるため、システム全体で見ると、一時的にキャッシュ上でデータが二重に保持され、キャッシュメモリの使用効率が低下する。特に、ストレージ乙側で受信した更新データを一定期間保持した後、ディスク装置に反映するような制御を行う場合、キャッシュメモリ使用効率の低下が著しいと予想される。

【００１０】また、ＵＳＰ５８３２２２で開示されている技術で、ＲＡＩＤレベル５のＲＡＩＤを構成する場合、更新データに対してパリティ(冗長データ)を生成するために必要なデータを、当該データに対応するパリティを格納しているストレージへ転送する必要がある。ＲＡＩＤレベル５のＲＡＩＤでは、一定の単位(ストライプ)毎にデータを分割し、複数のディスク装置に格納し(ストライピング)、一列のデータストライプに対して、１つ以上のパリティを生成し、データとは別のディスク装置へ格納する。パリティを格納するディスク装置はストライプ列毎に変えることで、データ更新時のパリティ

更新による特定ディスク装置への負荷集中を回避する。例えば、ＲＡＩＤが４台のディスク装置で構成され、データストライプＡ、Ｂ、Ｃに対して、パリティＰが保持されている場合に、データストライプＡに対してデータ更新が行われた場合を考える。このとき、データストライプＡ、Ｂを格納する２台のディスク装置はストレージ甲に、データストライプＣとパリティＰを格納する２台のディスク装置はストレージ乙に搭載されているとする。データストライプＡに更新があると、パリティＰの更新値作成に必要なデータ、すなわち、データストライプＡの更新前の値および更新値か、またはデータストライプＡの更新値およびデータストライプＢの更新前の値を、ストレージ甲から乙へ転送する必要がある。このため、ＲＡＩＤレベル１のＲＡＩＤの場合と同様、キャッシュメモリ使用効率の低下をまねいてしまう。

【００１１】本発明の目的は、クラスタ構成ストレージにおいてクラスタ間での負荷分散を実現することである。

【００１２】本発明の別の目的は、クラスタ間負荷分散を行う際にクラスタ間のデータ転送量を削減することで、システム性能の向上を図ることである。

【００１３】

【課題を解決するための手段】クラスタ構成ストレージでは、異なるクラスタのキャッシュメモリや制御情報を格納する制御用メモリへアクセスが可能なため、クラスタ間の緊密な連携を行うことができる。この特徴を生かし、本発明では、クラスタ構成ストレージの複数クラスタ配下のディスク装置間でＲＡＩＤを構成する。

【００１４】まず、構成されたＲＡＩＤがＲＡＩＤレベル１であり、ホストからのデータ更新はホスト要求に同期して実行されるものとする。ＲＡＩＤ内のあるデータが更新されると、当該データは、当該データを格納する第一のディスク装置を搭載した第一のクラスタのキャッシュメモリへ保持される。そして、当該データは、第一のディスク装置と当該データのミラー(冗長データ)ディスクである第二のクラスタの第二のディスク装置に書き込まれる。このとき第二のディスク装置への書き込みにおいては、第一のクラスタより第二のクラスタに対して、当該データの第二のディスク装置への書き込み要求が送信される。そして、第二のクラスタにより第一のクラスタのキャッシュメモリ上の当該データを用いて書き込み処理が行われる。

【００１５】また、構成されたＲＡＩＤがＲＡＩＤレベル５であり、ホストからのデータ更新は更新データをキャッシュメモリに格納した時点で完了し、ホスト要求とは非同期にディスク装置へ反映されるものとする。キャッシュメモリに格納された更新データがディスク反映対象に選ばれると、当該データのパリティ生成に必要なデータがキャッシュメモリに読み上げられる。具体的には、当該データの更新前値およびパリティの更新前値、もしくは当該データ以外の同ストライプ列データの更新前値が、各データの属するクラスタのキャッシュメモリへ格納される。読み上げ対象データが当該データが属するクラスタとは別クラスタに属する場合、別クラスタ側へ別クラスタのキャッシュメモリへの対象データの読み上げ要求を送信し、別クラスタ側で読み上げ処理が行われる。必要なデータが各クラスタのキャッシュメモリ上に準備できたら、排他的論理和演算によりパリティを生成する。この排他論理和演算はプロセッサで実行してもよいし、演算用の専用ハードウェアを搭載しても構わない。このとき、排他論理和演算対象の各データを演算実行ユニット(プロセッサか専用ハード)へ転送する必要があるが、クラスタ間のデータ転送量が最小となるようにパリティ生成実行クラスタを決定する。

【００１６】以上のような手段により、クラスタ構成ストレージにおいて、クラスタ間でのＲＡＩＤ構成を可能とし、クラスタ間での負荷分散を実現できる。

【００１７】また、クラスタ間での負荷分散実行時に、クラスタ間のデータ転送量を抑えて、ストレージのアクセス性能を向上させることが可能となる。

【００１８】

【発明の実施の形態】以下、本発明の実施形態について説明する。実施形態では、クラスタ構成ストレージにおいてクラスタ間で構成したＲＡＩＤにデータ更新が発生した場合の処理を例に説明する。なお、説明の簡略化のためにクラスタ数を２とするが、３クラスタ以上のディスク装置を用いてＲＡＩＤを構成しても構わない。また、本実施形態ではＲＡＩＤレベルは１と５を用いる。ＲＡＩＤレベル１は８台のディスク装置で構成し、ミラーディスク側にデータの複製を保持する。ＲＡＩＤレベル５も８台のディスク装置で構成し、各ストライプ列毎に７つのデータストライプと１つのパリティで構成されるものとする。なお、ＲＡＩＤの構成ディスク数、ＲＡＩＤレベル５のパリティ数はこれ以外の値であって構わない。

【００１９】また、説明の簡略化のため、ホストからのデータアクセス単位をＲＡＩＤレベル５のＲＡＩＤのストライプサイズと同じとしているが、現実には両者のサイズは必ずしも一致しない。その場合、ストライプに満たないデータが更新されたり、複数ストライプにまたがるデータ更新が発生する場合があるが、それぞれの場合のパリティ生成方法は従来公知の技術であり、本明細書では詳細は述べない。

【００２０】本発明の実施形態は第１から第２の実施形態がある。第１の実施形態はクラスタ間でＲＡＩＤレベル１のＲＡＩＤを構成し、更新データはホストからのライト要求と同期してディスク装置に反映する場合を示す。第２の実施形態はクラスタ間でＲＡＩＤレベル５のＲＡＩＤを構成、更新データはホスト要求とは非同期にディスク装置に反映する場合を示す。

更新による特定ディスク装置への負荷集中を回避する。例えば、ＲＡＩＤが４台のディスク装置で構成され、データストライプＡ、Ｂ、Ｃに対して、パリティＰが保持されている場合に、データストライプＡに対してデータ更新が行われた場合を考える。このとき、データストライプＡ、Ｂを格納する２台のディスク装置はストレージ甲に、データストライプＣとパリティＰを格納する２台のディスク装置はストレージ乙に搭載されているとする。データストライプＡに更新があると、パリティＰの更新値作成に必要なデータ、すなわち、データストライプＡの更新前の値および更新値か、またはデータストライプＡの更新値およびデータストライプＢの更新前の値を、ストレージ甲から乙へ転送する必要がある。このため、ＲＡＩＤレベル１のＲＡＩＤの場合と同様、キャッシュメモリ使用効率の低下をまねいてしまう。

【００１１】本発明の目的は、クラスタ構成ストレージにおいてクラスタ間での負荷分散を実現することである。

【００１２】本発明の別の目的は、クラスタ間負荷分散を行う際にクラスタ間のデータ転送量を削減することで、システム性能の向上を図ることである。

【００１３】

【課題を解決するための手段】クラスタ構成ストレージでは、異なるクラスタのキャッシュメモリや制御情報を格納する制御用メモリへアクセスが可能なため、クラスタ間の緊密な連携を行うことができる。この特徴を生かし、本発明では、クラスタ構成ストレージの複数クラスタ配下のディスク装置間でＲＡＩＤを構成する。

【００１４】まず、構成されたＲＡＩＤがＲＡＩＤレベル１であり、ホストからのデータ更新はホスト要求に同期して実行されるものとする。ＲＡＩＤ内のあるデータが更新されると、当該データは、当該データを格納する第一のディスク装置を搭載した第一のクラスタのキャッシュメモリへ保持される。そして、当該データは、第一のディスク装置と当該データのミラー(冗長データ)ディスクである第二のクラスタの第二のディスク装置に書き込まれる。このとき第二のディスク装置への書き込みにおいては、第一のクラスタより第二のクラスタに対して、当該データの第二のディスク装置への書き込み要求が送信される。そして、第二のクラスタにより第一のクラスタのキャッシュメモリ上の当該データを用いて書き込み処理が行われる。

【００１５】また、構成されたＲＡＩＤがＲＡＩＤレベル５であり、ホストからのデータ更新は更新データをキャッシュメモリに格納した時点で完了し、ホスト要求とは非同期にディスク装置へ反映されるものとする。キャッシュメモリに格納された更新データがディスク反映対象に選ばれると、当該データのパリティ生成に必要なデータがキャッシュメモリに読み上げられる。具体的には、当該データの更新前値およびパリティの更新前値、もしくは当該データ以外の同ストライプ列データの更新前値が、各データの属するクラスタのキャッシュメモリへ格納される。読み上げ対象データが当該データが属するクラスタとは別クラスタに属する場合、別クラスタ側へ別クラスタのキャッシュメモリへの対象データの読み上げ要求を送信し、別クラスタ側で読み上げ処理が行われる。必要なデータが各クラスタのキャッシュメモリ上に準備できたら、排他的論理和演算によりパリティを生成する。この排他論理和演算はプロセッサで実行してもよいし、演算用の専用ハードウェアを搭載しても構わない。このとき、排他論理和演算対象の各データを演算実行ユニット(プロセッサか専用ハード)へ転送する必要があるが、クラスタ間のデータ転送量が最小となるようにパリティ生成実行クラスタを決定する。

【００１６】以上のような手段により、クラスタ構成ストレージにおいて、クラスタ間でのＲＡＩＤ構成を可能とし、クラスタ間での負荷分散を実現できる。

【００１７】また、クラスタ間での負荷分散実行時に、クラスタ間のデータ転送量を抑えて、ストレージのアクセス性能を向上させることが可能となる。

【００１８】

【発明の実施の形態】以下、本発明の実施形態について説明する。実施形態では、クラスタ構成ストレージにおいてクラスタ間で構成したＲＡＩＤにデータ更新が発生した場合の処理を例に説明する。なお、説明の簡略化のためにクラスタ数を２とするが、３クラスタ以上のディスク装置を用いてＲＡＩＤを構成しても構わない。また、本実施形態ではＲＡＩＤレベルは１と５を用いる。ＲＡＩＤレベル１は８台のディスク装置で構成し、ミラーディスク側にデータの複製を保持する。ＲＡＩＤレベル５も８台のディスク装置で構成し、各ストライプ列毎に７つのデータストライプと１つのパリティで構成されるものとする。なお、ＲＡＩＤの構成ディスク数、ＲＡＩＤレベル５のパリティ数はこれ以外の値であって構わない。

【００１９】また、説明の簡略化のため、ホストからのデータアクセス単位をＲＡＩＤレベル５のＲＡＩＤのストライプサイズと同じとしているが、現実には両者のサイズは必ずしも一致しない。その場合、ストライプに満たないデータが更新されたり、複数ストライプにまたがるデータ更新が発生する場合があるが、それぞれの場合のパリティ生成方法は従来公知の技術であり、本明細書では詳細は述べない。

【００２０】本発明の実施形態は第１から第２の実施形態がある。第１の実施形態はクラスタ間でＲＡＩＤレベル１のＲＡＩＤを構成し、更新データはホストからのライト要求と同期してディスク装置に反映する場合を示す。第２の実施形態はクラスタ間でＲＡＩＤレベル５のＲＡＩＤを構成、更新データはホスト要求とは非同期にディスク装置に反映する場合を示す。

【００２１】まず第一に、図１から図５を参照して、第１の実施形態を説明する。

【００２２】図１は本発明の第１の実施形態の対象となる計算機システムの構成を示すブロック図である。

【００２３】１台のホストコンピュータ１００がチャネル１０１を介して２クラスタ１０２からなるストレージシステムに接続している。ホストコンピュータから見てストレージシステムは１台のストレージであり、２本のチャネル１０１のどちらからでもストレージシステム内の任意のデータにアクセスすることが可能である。

【００２４】ストレージシステムを構成する２つのクラスタ１０２は、各々が従来のストレージシステムに当る。独立したストレージシステムと同様に、クラスタ１０２内部には、１つ以上のチャネルコントローラ１０３や1つ以上のディスクコントローラ１０４、キャッシュメモリ１０６、共用メモリ１０７、パリティ生成器１０８を実装し、各コンポーネントは内部ネットワークにより互いに交信可能である。また、ディスクコントローラ１０４には複数台のディスク装置１０５が接続される。これらのコンポーネントおよび内部ネットワークや電源などは可用性向上のため多重化することが望ましい。さらに、クラスタ１０２間には、双方のコントローラから互いのキャッシュメモリ１０６や共用メモリ１０７がアクセスできるよう、クラスタ間ネットワーク１０９で接続されている。

【００２５】ホストコンピュータ１００では、各種アプリケーションプログラムが動作し、その実行に伴いストレージシステムへのデータアクセスを要求する。このとき、ホストコンピュータ１００はストレージシステムが提供する論理ディスクに対してアクセスを行う。論理ディスクはストレージシステムがホストコンピュータ１００に提供する見かけ上の記憶媒体で、ストレージシステム内で実際のディスク装置１０５への格納場所、格納方法を管理している。

【００２６】チャネルコントローラ１０３は、ホストコンピュータ１００から論理ディスクに対するアクセス要求を受け取り、各要求に見合ったディスク装置１０５を特定し、当該ディスク装置１０５へのリード/ライト要求をディスクコントローラ１０４へ送信する。

【００２７】ディスクコントローラ１０４は、チャネルコントローラ１０４からのリード/ライト要求に応じて、ディスク装置１０５へアクセスする。リード要求時は対象データまたはパリティをディスク装置１０５からキャッシュメモリ１０６へ読み上げ、ライト要求時はキャッシュメモリ１０６に格納された更新データまたは更新パリティを対応するディスク装置１０５へ書き出す。また、ディスク装置１０５への書き込みをホストコンピュータ１００からのライト要求とは非同期に実行する第２の実施例では、ディスクコントローラ１０４はキャッシュメモリ１０６に保持した複数の更新データのディスク装置１０５への反映スケジュールを決定する。すなわち、キャッシュメモリ１０６における更新データの占有率や、各更新データの滞留時間などを考慮して、周期的にディスク装置１０５への書き出し要否の判定、および書き出し対象データの決定を実施する。対象と決定された更新データはディスクコントローラ１０４にて処理されディスク装置１０５へ書き出される。

【００２８】キャッシュメモリ１０６は、ホストコンピュータ１００とディスク装置１０５との間の転送を仲介する記憶媒体であり、保持されるデータはチャネルコントローラ１０３およびディスクコントローラ１０４にて協同管理される。ホストコンピュータ１００からのライトデータは一旦キャッシュメモリ１０６に格納され、然る後ディスク装置１０５へ書き出される。逆にホストコンピュータ１００からのリード要求に対して、ディスク装置１０５から当該データを一旦キャッシュメモリ１０６へ読み上げ、然る後、ホストコンピュータ１００へ転送される。このとき、ホストコンピュータ１００は論理ディスクに対してアクセスするため、キャッシュメモリ１０６上では論理ディスク内のアドレスによりデータを管理し、ディスク装置へのアクセス時には、当該データの対応するディスク装置１０５、および当該ディスク装置１０５内のアドレスを特定する必要がある。

【００２９】キャッシュメモリ１０６の管理方法としては様々な方式が考えられ、現在様々な方式が各社製品に採用されている。管理方法の一例としては、キャッシュメモリ１０６を特定サイズ毎に分割し、このデータ単位毎に管理する方法がある。データ単位としては、例えばＲＡＩＤのストライプサイズなどが適している。このデータ単位を便宜上セグメントと呼ぶ。全キャッシュ領域はセグメント単位に管理され、最初は全セグメントが未割当ての状態である。ホストコンピュータ１００から論理ディスクの特定領域にアクセスが生じた場合、特定領域の対応するデータストライプに対して未割当てのセグメントの一つを割当て、ライト/リードデータを格納する。セグメントとデータストライプ間の対応はキャッシュ管理情報として管理され、共用メモリ１０７上に保持される。

【００３０】本実施形態では、キャッシュメモリ１０６上でのデータ保持は、当該データセグメントが格納されるディスク装置１０５を搭載するクラスタ１０２のキャッシュメモリ１０６に一元的に保持するものとする。この場合、アクセスデータにセグメントを割当てる処理において、当該データストライプが属するクラスタを算出する必要がある。別の方法としては、論理ディスク毎に使用するキャッシュメモリ１０６を決める方法も考えられる。このような管理の場合、各論理ディスク毎に対応するディスク装置１０５の所属クラスタ１０２などを考慮して、当該論理ディスクの所属クラスタ１０２を決定し、対応情報を共用メモリ１０７に保持する必要があ

【００２１】まず第一に、図１から図５を参照して、第１の実施形態を説明する。

【００２２】図１は本発明の第１の実施形態の対象となる計算機システムの構成を示すブロック図である。

【００２３】１台のホストコンピュータ１００がチャネル１０１を介して２クラスタ１０２からなるストレージシステムに接続している。ホストコンピュータから見てストレージシステムは１台のストレージであり、２本のチャネル１０１のどちらからでもストレージシステム内の任意のデータにアクセスすることが可能である。

【００２４】ストレージシステムを構成する２つのクラスタ１０２は、各々が従来のストレージシステムに当る。独立したストレージシステムと同様に、クラスタ１０２内部には、１つ以上のチャネルコントローラ１０３や1つ以上のディスクコントローラ１０４、キャッシュメモリ１０６、共用メモリ１０７、パリティ生成器１０８を実装し、各コンポーネントは内部ネットワークにより互いに交信可能である。また、ディスクコントローラ１０４には複数台のディスク装置１０５が接続される。これらのコンポーネントおよび内部ネットワークや電源などは可用性向上のため多重化することが望ましい。さらに、クラスタ１０２間には、双方のコントローラから互いのキャッシュメモリ１０６や共用メモリ１０７がアクセスできるよう、クラスタ間ネットワーク１０９で接続されている。

【００２５】ホストコンピュータ１００では、各種アプリケーションプログラムが動作し、その実行に伴いストレージシステムへのデータアクセスを要求する。このとき、ホストコンピュータ１００はストレージシステムが提供する論理ディスクに対してアクセスを行う。論理ディスクはストレージシステムがホストコンピュータ１００に提供する見かけ上の記憶媒体で、ストレージシステム内で実際のディスク装置１０５への格納場所、格納方法を管理している。

【００２６】チャネルコントローラ１０３は、ホストコンピュータ１００から論理ディスクに対するアクセス要求を受け取り、各要求に見合ったディスク装置１０５を特定し、当該ディスク装置１０５へのリード/ライト要求をディスクコントローラ１０４へ送信する。

【００２７】ディスクコントローラ１０４は、チャネルコントローラ１０４からのリード/ライト要求に応じて、ディスク装置１０５へアクセスする。リード要求時は対象データまたはパリティをディスク装置１０５からキャッシュメモリ１０６へ読み上げ、ライト要求時はキャッシュメモリ１０６に格納された更新データまたは更新パリティを対応するディスク装置１０５へ書き出す。また、ディスク装置１０５への書き込みをホストコンピュータ１００からのライト要求とは非同期に実行する第２の実施例では、ディスクコントローラ１０４はキャッシュメモリ１０６に保持した複数の更新データのディスク装置１０５への反映スケジュールを決定する。すなわち、キャッシュメモリ１０６における更新データの占有率や、各更新データの滞留時間などを考慮して、周期的にディスク装置１０５への書き出し要否の判定、および書き出し対象データの決定を実施する。対象と決定された更新データはディスクコントローラ１０４にて処理されディスク装置１０５へ書き出される。

【００２８】キャッシュメモリ１０６は、ホストコンピュータ１００とディスク装置１０５との間の転送を仲介する記憶媒体であり、保持されるデータはチャネルコントローラ１０３およびディスクコントローラ１０４にて協同管理される。ホストコンピュータ１００からのライトデータは一旦キャッシュメモリ１０６に格納され、然る後ディスク装置１０５へ書き出される。逆にホストコンピュータ１００からのリード要求に対して、ディスク装置１０５から当該データを一旦キャッシュメモリ１０６へ読み上げ、然る後、ホストコンピュータ１００へ転送される。このとき、ホストコンピュータ１００は論理ディスクに対してアクセスするため、キャッシュメモリ１０６上では論理ディスク内のアドレスによりデータを管理し、ディスク装置へのアクセス時には、当該データの対応するディスク装置１０５、および当該ディスク装置１０５内のアドレスを特定する必要がある。

【００２９】キャッシュメモリ１０６の管理方法としては様々な方式が考えられ、現在様々な方式が各社製品に採用されている。管理方法の一例としては、キャッシュメモリ１０６を特定サイズ毎に分割し、このデータ単位毎に管理する方法がある。データ単位としては、例えばＲＡＩＤのストライプサイズなどが適している。このデータ単位を便宜上セグメントと呼ぶ。全キャッシュ領域はセグメント単位に管理され、最初は全セグメントが未割当ての状態である。ホストコンピュータ１００から論理ディスクの特定領域にアクセスが生じた場合、特定領域の対応するデータストライプに対して未割当てのセグメントの一つを割当て、ライト/リードデータを格納する。セグメントとデータストライプ間の対応はキャッシュ管理情報として管理され、共用メモリ１０７上に保持される。

【００３０】本実施形態では、キャッシュメモリ１０６上でのデータ保持は、当該データセグメントが格納されるディスク装置１０５を搭載するクラスタ１０２のキャッシュメモリ１０６に一元的に保持するものとする。この場合、アクセスデータにセグメントを割当てる処理において、当該データストライプが属するクラスタを算出する必要がある。別の方法としては、論理ディスク毎に使用するキャッシュメモリ１０６を決める方法も考えられる。このような管理の場合、各論理ディスク毎に対応するディスク装置１０５の所属クラスタ１０２などを考慮して、当該論理ディスクの所属クラスタ１０２を決定し、対応情報を共用メモリ１０７に保持する必要があ

る。

【００３１】各セグメントは、格納データがディスク装置１０５へ反映済みか否かで区別され管理される。前者をクリーン状態、後者をダーティ状態と呼ぶ。また、各セグメント内の情報が有効であるかを示す情報もキャッシュ制御情報として保持する。例えば、リード用に新規に割当てられたセグメントはその時点ではクリーン状態であるが、ディスク装置１０５からのデータ読み込みが完了しない限り、内部に保持するデータは無効である。また、セグメントの割当てについては、仮に新規にセグメントが必要なときに未割当てのセグメントが無い場合には、クリーンセグメントの一つを転用する。クリーンセグメントも存在しない場合は、ダーティセグメントの未反映データをディスク装置１０５へ反映後、転用する。さらに、更新データがＲＡＩＤに属する場合、ディスク装置１０５への未反映状態には二つの状態が存在する。ＲＡＩＤレベル１のＲＡＩＤの場合、データディスクへもミラーディスクへも未反映な状態、ミラーディスクへのみ反映済みの状態がある。ＲＡＩＤレベル５のＲＡＩＤの場合、パリティ未生成の状態、パリティ生成済みの状態がある。本実施形態では、ホストコンピュータ１００からキャッシュメモリ１０６へ書き込まれた状態をホストダーティ状態、ＲＡＩＤレベル１のミラーディスクへ反映済み状態またはＲＡＩＤレベル５のパリティ生成済み状態を物理ダーティ状態と呼ぶことにする。なお、ＲＡＩＤレベル１のディスク装置１０５への反映順序はミラーディスク、データディスクの順とするが、この順序が逆であっても構わない。また、各更新データ毎に任意の順序でデータディスク又はミラーディスクへ反映しても構わない。ただし、その場合、データディスクとミラーディスクそれぞれ独立にダーティ状態を管理する必要がある。また、ＲＡＩＤレベル５の場合、更新データに対する更新前のデータをキャッシュメモリ１０６上へ読み上げる必要があるため、同一データストライプに対してニ値を保持する必要がある。このためには、各データストライプについて更新データと更新前のデータを管理する最大２つのセグメントを割当てられるよう制御し、これら２つのセグメントを同一のデータストライプに対応づけて管理する。

【００３２】共用メモリ１０７は、チャネルコントローラ１０３やディスクコントローラ１０４がＩ／Ｏ制御を行うのに必要な制御情報を保持している。制御情報の例としては、各コントローラで動作する処理単位であるジョブの管理情報や、ディスク装置１０５の管理情報、キャッシュメモリ１０４上でのデータ管理情報などが挙げられる。また、先述した論理ディスクとディスク装置１０５との対応情報も保持している。

【００３３】図３に論理ディスクとディスク装置１０５との対応情報の例を示す。対応情報は各論理ディスク毎のエントリを持ち、各論理ディスク毎にＲＡＩＤ情報、構成ディスク装置リストなどで構成される。ＲＡＩＤ情報はＲＡＩＤレベル、ストライプサイズ、データストライプ数、パリティストライプ数、そしてパリティ格納ディスク装置がいくつのストライプ列毎に変わるかを示すパリティサイクルからなる。構成ディスク装置リストは、各構成ディスク装置のクラスタ番号とクラスタ内ディスク番号のリストからなる。

【００３４】パリティ生成器１０８は、ＲＡＩＤレベル５のＲＡＩＤにおいてパリティを演算するのに用いられる。パリティを生成するのに必要な情報とは、更新データに対する更新値と更新前値および対応するパリティの更新前値、もしくは更新データの更新値および同じストライプ列の他の全てのデータの更新前値である。本実施形態では、前者の情報を用いたパリティ生成で説明する。必要な情報をパリティ生成器に入力し、排他的論理和演算を行い、演算結果であるパリティの更新値をキャッシュメモリ１０６に格納する。

【００３５】ディスク装置１０５は、ホストコンピュータ１００から見た見掛けのディスク装置である論理ディスクのデータを格納する。論理ディスクとディスク装置１０５の対応は図３に示す情報で管理され、論理ディスクへのアクセスに対して当該制御情報を用いてディスク装置１０５を算出して、対応するディスク装置１０５へアクセスが行われる。

【００３６】クラスタ間ネットワーク１０９は、２つのクラスタ間を接続するネットワークであり、両クラスタの各コントローラは互いのキャッシュメモリ１０６および共用メモリ１０７へアクセスすることが可能である。ただ、通常、クラスタ内で各コンポーネントを接続する内部ネットワークと比較すれば、転送能力は低く、ストレージシステムの性能向上には、極力クラスタ間のデータ交信量を抑制する必要がある。

【００３７】次に第１の実施形態におけるデータ更新時の処理の流れについて説明する。

【００３８】図２は第１の実施形態においてクラスタ間にまたがるＲＡＩＤに対してデータ更新が行われた場合の処理の流れを示すものである。２つのクラスタで構成されるストレージにおいて、両クラスタに搭載されたディスク装置４台ずつ、計８台でＲＡＩＤレベル１のＲＡＩＤを構成する。当該ＲＡＩＤデータへの更新を受けた一方のストレージは当該データが属する第一のクラスタのキャッシュメモリに当該更新データを格納し、ミラーディスク側の第二のクラスタへ当該更新データのディスク装置への書き込み要求を送信する。当該要求を受けた第二のクラスタでは、第一のクラスタのキャッシュメモリ上の当該更新データを用いて、ミラーディスクに当る第二のディスク装置へ書き込みを行う。その後、当該更新データを格納先である第一のディスク装置に書き込み、ホストへライト処理の完了を報告する。

【００３９】次に第１の実施形態における各処理につい

る。

【００３１】各セグメントは、格納データがディスク装置１０５へ反映済みか否かで区別され管理される。前者をクリーン状態、後者をダーティ状態と呼ぶ。また、各セグメント内の情報が有効であるかを示す情報もキャッシュ制御情報として保持する。例えば、リード用に新規に割当てられたセグメントはその時点ではクリーン状態であるが、ディスク装置１０５からのデータ読み込みが完了しない限り、内部に保持するデータは無効である。また、セグメントの割当てについては、仮に新規にセグメントが必要なときに未割当てのセグメントが無い場合には、クリーンセグメントの一つを転用する。クリーンセグメントも存在しない場合は、ダーティセグメントの未反映データをディスク装置１０５へ反映後、転用する。さらに、更新データがＲＡＩＤに属する場合、ディスク装置１０５への未反映状態には二つの状態が存在する。ＲＡＩＤレベル１のＲＡＩＤの場合、データディスクへもミラーディスクへも未反映な状態、ミラーディスクへのみ反映済みの状態がある。ＲＡＩＤレベル５のＲＡＩＤの場合、パリティ未生成の状態、パリティ生成済みの状態がある。本実施形態では、ホストコンピュータ１００からキャッシュメモリ１０６へ書き込まれた状態をホストダーティ状態、ＲＡＩＤレベル１のミラーディスクへ反映済み状態またはＲＡＩＤレベル５のパリティ生成済み状態を物理ダーティ状態と呼ぶことにする。なお、ＲＡＩＤレベル１のディスク装置１０５への反映順序はミラーディスク、データディスクの順とするが、この順序が逆であっても構わない。また、各更新データ毎に任意の順序でデータディスク又はミラーディスクへ反映しても構わない。ただし、その場合、データディスクとミラーディスクそれぞれ独立にダーティ状態を管理する必要がある。また、ＲＡＩＤレベル５の場合、更新データに対する更新前のデータをキャッシュメモリ１０６上へ読み上げる必要があるため、同一データストライプに対して二値を保持する必要がある。このためには、各データストライプについて更新データと更新前のデータを管理する最大２つのセグメントを割当てられるよう制御し、これら２つのセグメントを同一のデータストライプに対応づけて管理する。

【００３２】共用メモリ１０７は、チャネルコントローラ１０３やディスクコントローラ１０４がＩ／Ｏ制御を行うのに必要な制御情報を保持している。制御情報の例としては、各コントローラで動作する処理単位であるジョブの管理情報や、ディスク装置１０５の管理情報、キャッシュメモリ１０４上でのデータ管理情報などが挙げられる。また、先述した論理ディスクとディスク装置１０５との対応情報も保持している。

【００３３】図３に論理ディスクとディスク装置１０５との対応情報の例を示す。対応情報は各論理ディスク毎のエントリを持ち、各論理ディスク毎にＲＡＩＤ情報、構成ディスク装置リストなどで構成される。ＲＡＩＤ情報はＲＡＩＤレベル、ストライプサイズ、データストライプ数、パリティストライプ数、そしてパリティ格納ディスク装置がいくつのストライプ列毎に変わるかを示すパリティサイクルからなる。構成ディスク装置リストは、各構成ディスク装置のクラスタ番号とクラスタ内ディスク番号のリストからなる。

【００３４】パリティ生成器１０８は、ＲＡＩＤレベル５のＲＡＩＤにおいてパリティを演算するのに用いられる。パリティを生成するのに必要な情報とは、更新データに対する更新値と更新前値および対応するパリティの更新前値、もしくは更新データの更新値および同じストライプ列の他の全てのデータの更新前値である。本実施形態では、前者の情報を用いたパリティ生成で説明する。必要な情報をパリティ生成器に入力し、排他的論理和演算を行い、演算結果であるパリティの更新値をキャッシュメモリ１０６に格納する。

【００３５】ディスク装置１０５は、ホストコンピュータ１００から見た見掛けのディスク装置である論理ディスクのデータを格納する。論理ディスクとディスク装置１０５の対応は図３に示す情報で管理され、論理ディスクへのアクセスに対して当該制御情報を用いてディスク装置１０５を算出して、対応するディスク装置１０５へアクセスが行われる。

【００３６】クラスタ間ネットワーク１０９は、２つのクラスタ間を接続するネットワークであり、両クラスタの各コントローラは互いのキャッシュメモリ１０６および共用メモリ１０７へアクセスすることが可能である。ただ、通常、クラスタ内で各コンポーネントを接続する内部ネットワークと比較すれば、転送能力は低く、ストレージシステムの性能向上には、極力クラスタ間のデータ交信量を抑制する必要がある。

【００３７】次に第１の実施形態におけるデータ更新時の処理の流れについて説明する。

【００３８】図２は第１の実施形態においてクラスタ間にまたがるＲＡＩＤに対してデータ更新が行われた場合の処理の流れを示すものである。２つのクラスタで構成されるストレージにおいて、両クラスタに搭載されたディスク装置４台ずつ、計８台でＲＡＩＤレベル１のＲＡＩＤを構成する。当該ＲＡＩＤデータへの更新を受けた一方のストレージは当該データが属する第一のクラスタのキャッシュメモリに当該更新データを格納し、ミラーディスク側の第二のクラスタへ当該更新データのディスク装置への書き込み要求を送信する。当該要求を受けた第二のクラスタでは、第一のクラスタのキャッシュメモリ上の当該更新データを用いて、ミラーディスクに当る第二のディスク装置へ書き込みを行う。その後、当該更新データを格納先である第一のディスク装置に書き込み、ホストへライト処理の完了を報告する。

【００３９】次に第１の実施形態における各処理につい

てフロー図を用いて詳細に説明する。

【００４０】図４はチャネルコントローラ１０３におけるチャネルコマンド処理の処理フロー図である。当処理はホストコンピュータ１００からのＩ／Ｏ要求を受けつけ、要求処理内容に応じて、ディスクコントローラ１０４に処理要求を送信する。

【００４１】ステップ４０１で、ホストアクセス対象のデータが格納されるディスク装置１０５の属するクラスタを特定する。クラスタの特定には、図３に示した論理ディスクとディスク装置１０５の対応情報を用いる。まず、論理ディスクの特定領域に対するアクセスを受信したら、対応情報を元に当該論理ディスクの当該領域がＲＡＩＤ内のいくつ目のディスク装置に格納されるかを算出する。それから、そのディスク装置がどのクラスタのどのディスク装置１０５であるかを対応情報内のディスク装置リストを参照して求める。ここで特定したクラスタを第１のクラスタ、ディスク装置を第１のディスク装置とする。

【００４２】ステップ４０２でホスト要求を判定し、リード要求であるならステップ４０９へ、ライト要求であるならステップ４０３へ遷移する。ホスト要求にはその他の要求も考えられるが、本実施形態では簡単のため省略している。

【００４３】ステップ４０３からステップ４０７はライト要求時の処理である。ステップ４０３では、アクセス対象のデータストライプに対してステップ４０１で特定した第一のクラスタ１０２のキャッシュメモリ１０６から空きセグメントを割当て、ホストからライトデータを受け取り、当該セグメントへ格納する。セグメントの割当て時には、対象となるクラスタ１０２の共用メモリ１０７に保持されているキャッシュ管理情報を参照/更新する。具体的には、第一のクラスタのキャッシュ管理情報のアクセス排他をかけた状態で、空きセグメントの管理情報から任意の空きセグメントを獲得し、当該セグメントを当該データストライプへ対応付け、セグメント状態を空き(未割当て)状態から割当て状態かつホストダーティ状態へ変更する。

【００４４】ステップ４０４では、当該データストライプに対応するミラーディスクがどのクラスタ１０２のどのディスク装置１０５であるかを特定する。ここで特定したクラスタを第２のクラスタ、ディスク装置を第２のディスク装置とする。

【００４５】ステップ４０５では、第２のクラスタのディスクコントローラ１０４に対して、第２のディスク装置に対する当該データストライプのライト要求を送信する。本実施形態では、コントローラ間の処理要求は要求内容を示す数バイト程度の情報を直接送信先クラスタ１０２の共用メモリ１０７へ書き込むことで行う。当該送信情報をＭＳＧと呼ぶことにする。ＭＳＧ内の情報としては、例えば処理要求種別、処理対象データ/パリティストライプアドレス情報など、場合によってはセグメントの特定情報や対応するディスク装置１０５アドレス情報を送信してもよい。ＭＳＧの管理には様々な方式が考えられるが、例えば共用メモリ１０７上に対象コントローラ単位、さらに要求処理単位にＭＳＧを管理するキューのようなものを設け、このキューにＭＳＧを登録する方式が考えられる。各コントローラは各自の担当するキューを周期的に監視することでＭＳＧを受信することができる。また、要求ＭＳＧの完了報告も同様に送信元へのＭＳＧとして送信することもできるし、あるいは送信元ジョブの管理情報内に直接処理結果を書き込んでも構わない。第２のクラスタのディスクコントローラ１０４からのライト処理完了報告を受けたら、ステップ４０６で第１のクラスタのディスクコントローラ１０５に対して、第１のディスク装置に対する当該データストライプのライト要求を送信する。ディスクコントローラ１０５からのライト完了報告を受けたら、ステップ４０７でホストコンピュータ１００に当該ライト要求の完了報告を行う。

【００４６】次に、ステップ４０８からステップ４１０でリード要求時の処理について説明する。

【００４７】ステップ４０８では、ステップ４０３と同様、リード対象データセグメントに対して、第１のクラスタのキャッシュメモリ１０６のセグメントを割当てる。ステップ４０３と異なるのは、割当てたセグメントはクリーン状態となる点である。

【００４８】ステップ４０９では、第１のクラスタのディスクコントローラ１０４に対して、第１のディスク装置に対する当該データストライプのリード要求を送信する。当該要求の完了報告を受け取ったら、ステップ４１０で対象セグメント内のデータをホストコンピュータ１００へ転送し、ステップ４０７で完了報告を行う。

【００４９】図５はディスクコントローラ１０４で動作するディスクコマンド処理の処理フロー図である。第１の実施形態では、当処理はチャネルコマンド処理より送信されるディスク装置１０５へのリード/ライト処理を受信し、処理する。

【００５０】まず、ステップ５０１で受信した処理要求ＭＳＧ内の処理要求種別を判定し、リード要求時はステップ５０６へ、ライト要求時はステップ５０２へそれぞれ遷移する。

【００５１】ステップ５０２では、処理要求ＭＳＧ内のデータストライプアドレスやセグメント状態などの整合性をチェックする。セグメント状態のチェックでは、例えばミラーディスクへの書き込みである場合は対象セグメントがホストダーティ状態であること、データディスクへの書き込みである場合は物理ダーティ状態であることを確認する。ＭＳＧ内にディスク装置１０５アドレスなどが含まれない場合には、図３で示した論理ディスクとディスク装置１０５の対応情報を用いてディスク装置

てフロー図を用いて詳細に説明する。

【００４０】図４はチャネルコントローラ１０３におけるチャネルコマンド処理の処理フロー図である。当処理はホストコンピュータ１００からのＩ／Ｏ要求を受けつけ、要求処理内容に応じて、ディスクコントローラ１０４に処理要求を送信する。

【００４１】ステップ４０１で、ホストアクセス対象のデータが格納されるディスク装置１０５の属するクラスタを特定する。クラスタの特定には、図３に示した論理ディスクとディスク装置１０５の対応情報を用いる。まず、論理ディスクの特定領域に対するアクセスを受信したら、対応情報を元に当該論理ディスクの当該領域がＲＡＩＤ内のいくつ目のディスク装置に格納されるかを算出する。それから、そのディスク装置がどのクラスタのどのディスク装置１０５であるかを対応情報内のディスク装置リストを参照して求める。ここで特定したクラスタを第１のクラスタ、ディスク装置を第１のディスク装置とする。

【００４２】ステップ４０２でホスト要求を判定し、リード要求であるならステップ４０９へ、ライト要求であるならステップ４０３へ遷移する。ホスト要求にはその他の要求も考えられるが、本実施形態では簡単のため省略している。

【００４３】ステップ４０３からステップ４０７はライト要求時の処理である。ステップ４０３では、アクセス対象のデータストライプに対してステップ４０１で特定した第一のクラスタ１０２のキャッシュメモリ１０６から空きセグメントを割当て、ホストからライトデータを受け取り、当該セグメントへ格納する。セグメントの割当て時には、対象となるクラスタ１０２の共用メモリ１０７に保持されているキャッシュ管理情報を参照/更新する。具体的には、第一のクラスタのキャッシュ管理情報のアクセス排他をかけた状態で、空きセグメントの管理情報から任意の空きセグメントを獲得し、当該セグメントを当該データストライプへ対応付け、セグメント状態を空き(未割当て)状態から割当て状態かつホストダーティ状態へ変更する。

【００４４】ステップ４０４では、当該データストライプに対応するミラーディスクがどのクラスタ１０２のどのディスク装置１０５であるかを特定する。ここで特定したクラスタを第２のクラスタ、ディスク装置を第２のディスク装置とする。

【００４５】ステップ４０５では、第２のクラスタのディスクコントローラ１０４に対して、第２のディスク装置に対する当該データストライプのライト要求を送信する。本実施形態では、コントローラ間の処理要求は要求内容を示す数バイト程度の情報を直接送信先クラスタ１０２の共用メモリ１０７へ書き込むことで行う。当該送信情報をＭＳＧと呼ぶことにする。ＭＳＧ内の情報としては、例えば処理要求種別、処理対象データ/パリティストライプアドレス情報など、場合によってはセグメントの特定情報や対応するディスク装置１０５アドレス情報を送信してもよい。ＭＳＧの管理には様々な方式が考えられるが、例えば共用メモリ１０７上に対象コントローラ単位、さらに要求処理単位にＭＳＧを管理するキューのようなものを設け、このキューにＭＳＧを登録する方式が考えられる。各コントローラは各自の担当するキューを周期的に監視することでＭＳＧを受信することができる。また、要求ＭＳＧの完了報告も同様に送信元へのＭＳＧとして送信することもできるし、あるいは送信元ジョブの管理情報内に直接処理結果を書き込んでも構わない。第２のクラスタのディスクコントローラ１０４からのライト処理完了報告を受けたら、ステップ４０６で第１のクラスタのディスクコントローラ１０５に対して、第１のディスク装置に対する当該データストライプのライト要求を送信する。ディスクコントローラ１０５からのライト完了報告を受けたら、ステップ４０７でホストコンピュータ１００に当該ライト要求の完了報告を行う。

【００４６】次に、ステップ４０８からステップ４１０でリード要求時の処理について説明する。

【００４７】ステップ４０８では、ステップ４０３と同様、リード対象データセグメントに対して、第１のクラスタのキャッシュメモリ１０６のセグメントを割当てる。ステップ４０３と異なるのは、割当てたセグメントはクリーン状態となる点である。

【００４８】ステップ４０９では、第１のクラスタのディスクコントローラ１０４に対して、第１のディスク装置に対する当該データストライプのリード要求を送信する。当該要求の完了報告を受け取ったら、ステップ４１０で対象セグメント内のデータをホストコンピュータ１００へ転送し、ステップ４０７で完了報告を行う。

【００４９】図５はディスクコントローラ１０４で動作するディスクコマンド処理の処理フロー図である。第１の実施形態では、当処理はチャネルコマンド処理より送信されるディスク装置１０５へのリード/ライト処理を受信し、処理する。

【００５０】まず、ステップ５０１で受信した処理要求ＭＳＧ内の処理要求種別を判定し、リード要求時はステップ５０６へ、ライト要求時はステップ５０２へそれぞれ遷移する。

【００５１】ステップ５０２では、処理要求ＭＳＧ内のデータストライプアドレスやセグメント状態などの整合性をチェックする。セグメント状態のチェックでは、例えばミラーディスクへの書き込みである場合は対象セグメントがホストダーティ状態であること、データディスクへの書き込みである場合は物理ダーティ状態であることを確認する。ＭＳＧ内にディスク装置１０５アドレスなどが含まれない場合には、図３で示した論理ディスクとディスク装置１０５の対応情報を用いてディスク装置

１０５を算出する。また、指定されたデータストライプのセグメントアドレスがＭＳＧに含まれない場合には、キャッシュ管理情報の対応情報を用いて、セグメントアドレスを特定する。

【００５２】ステップ５０３では、対象となるディスク装置１０５へ対象セグメント内データを書き込む。書き込みが正常に完了したら、当該セグメントのキャッシュ管理情報を更新する。具体的には、当該ライトがミラーディスクに対するライトの場合は当該セグメントを物理ダーティ状態に、データディスクに対するライトの場合は当該セグメントをクリーン状態にそれぞれ変更する。これらの処理が完了したら、ステップ５０５で当該処理の要求元へ処理の完了を報告する。

【００５３】次にステップ５０６からステップ５０８を用いてリード処理について説明する。

【００５４】ステップ５０６では、ステップ５０２と同様、処理要求ＭＳＧのアドレスやセグメント状態などの整合性チェックを行う。リード要求の場合、対象となるセグメントが割当てられていてかつ、クリーン状態である、もしくはダーティ状態でかつ更新前データ/パリティ用のセグメントが確保されているかを確認する。ダーティ状態でかつ更新前データ用セグメントが割当てられていない場合は上位にエラー報告してもよいし、当該処理にて当該データストライプにセグメントを追加割当てしても構わない。

【００５５】ステップ５０７では、対象となるディスク装置１０５から当該セグメントへ対象データを読み上げ、データが読み上げられたらステップ５０８で当該セグメント内のデータが有効であるよう制御情報を変更する。全ての処理が完了したらライト要求と同様にステップ５０５で上位へ要求処理の完了を報告する。

【００５６】第二に、図６から図１０を参照して、第２の実施形態を説明する。

【００５７】図６に第２の実施形態においてクラスタ間にまたがるＲＡＩＤに対してデータ更新が行われた場合の処理の流れを示す。２つのクラスタで構成されるストレージにおいて、両クラスタに搭載されたディスク装置４台ずつ、計８台でＲＡＩＤレベル５のＲＡＩＤを構築する。当該ＲＡＩＤデータへの更新を受けた一方のクラスタは当該データが属する第一のクラスタのキャッシュメモリに当該更新データを格納し、ホストへライト処理の完了を報告する。なお、本図ではホスト要求を受けたクラスタと当該データが属する第一のクラスタが同一である場合を想定している。当該更新データは、キャッシュメモリ上での更新データの保持数などを考慮しながら非同期にディスク装置へ書き込まれる。当該更新データがディスク装置への反映対象となった場合、対応するパリティを生成するために必要なデータをディスク装置から対応するクラスタのキャッシュメモリ１０６へそれぞれ読み上げる。簡単のため、更新されたデータストライプの更新前の値と、対応するパリティの更新前の値を読み上げるものとする。このとき、パリティがデータストライプが属する第一のクラスタでなく第二のクラスタに属する場合、第二のクラスタに対して第二のクラスタのキャッシュメモリへパリティの更新前値を読み上げる要求を送信する。当該要求を受けた第二のクラスタでは、キャッシュメモリへパリティの更新前値を読み上げる。これと並行して第一のクラスタでは当該キャッシュメモリに当該更新データストライプの更新前値が読み上げられる。必要なデータが全て揃ったら、データおよびパリティの配置から最適なパリティ生成クラスタを決定し、そのクラスタにてパリティ更新値を生成する。その後、当該更新データおよび当該更新パリティを各々が属するクラスタにより各ディスク装置へ書き込む。

【００５８】次に第２の実施形態における各処理についてフロー図を用いて詳細に説明する。

【００５９】図７はチャネルコントローラ１０３上で動作するチャネルコマンド処理の処理フロー図である。処理フローは第１の実施形態のチャネルコマンド処理のフロー図である図４と共通部分が多い。よって、相違部分であるライト処理部分について説明する。

【００６０】ステップ７０２でホスト要求がライト要求であると判定したら、ステップ７０３でステップ４０３と同様に対象クラスタのキャッシュメモリ１０６からセグメントを割当て、ライトデータを格納する。当該セグメントをダーティ状態にし、格納するデータが有効である旨、制御情報を変更したら、その時点でホストコンピュータ１００へ要求処理の完了を報告する。このとき、データ消失を避けるためには、キャッシュメモリ１０６はバッテリを用いるなどの手段により不揮発化する必要がある。

【００６１】このようにホストコンピュータ１００からの更新データはキャッシュメモリ１０６に保持され、以降、ディスクコントローラ１０４の非同期ディスク反映処理により、ホスト要求とは非同期にディスク装置１０５へ反映される。

【００６２】図８はディスクコントローラ１０４上で動作する非同期ディスク反映処理の処理フロー図である。本処理はキャッシュメモリ１０６内の更新データ量やディスクコントローラ１０４の負荷を考慮しながら起動要否を判定され、更新されてからの経過時間などを元に対象となるデータを選出され起動される。この起動要否判定、および対象データの選出は既存論理であるため、詳細説明は省略する。

【００６３】ステップ８０１で、処理対象である更新データストライプのパリティが所属するクラスタを特定する。クラスタ特定には図３で示した論理ディスクとディスク装置１０５の対応情報を用いる。本実施形態では、パリティが所属クラスタはデータの所属クラスタとは異なる場合について説明する。以後、パリティ所属クラス

１０５を算出する。また、指定されたデータストライプのセグメントアドレスがＭＳＧに含まれない場合には、キャッシュ管理情報の対応情報を用いて、セグメントアドレスを特定する。

【００５２】ステップ５０３では、対象となるディスク装置１０５へ対象セグメント内データを書き込む。書き込みが正常に完了したら、当該セグメントのキャッシュ管理情報を更新する。具体的には、当該ライトがミラーディスクに対するライトの場合は当該セグメントを物理ダーティ状態に、データディスクに対するライトの場合は当該セグメントをクリーン状態にそれぞれ変更する。これらの処理が完了したら、ステップ５０５で当該処理の要求元へ処理の完了を報告する。

【００５３】次にステップ５０６からステップ５０８を用いてリード処理について説明する。

【００５４】ステップ５０６では、ステップ５０２と同様、処理要求ＭＳＧのアドレスやセグメント状態などの整合性チェックを行う。リード要求の場合、対象となるセグメントが割当てられていてかつ、クリーン状態である、もしくはダーティ状態でかつ更新前データ/パリティ用のセグメントが確保されているかを確認する。ダーティ状態でかつ更新前データ用セグメントが割当てられていない場合は上位にエラー報告してもよいし、当該処理にて当該データストライプにセグメントを追加割当てしても構わない。

【００５５】ステップ５０７では、対象となるディスク装置１０５から当該セグメントへ対象データを読み上げ、データが読み上げられたらステップ５０８で当該セグメント内のデータが有効であるよう制御情報を変更する。全ての処理が完了したらライト要求と同様にステップ５０５で上位へ要求処理の完了を報告する。

【００５６】第二に、図６から図１０を参照して、第２の実施形態を説明する。

【００５７】図６に第２の実施形態においてクラスタ間にまたがるＲＡＩＤに対してデータ更新が行われた場合の処理の流れを示す。２つのクラスタで構成されるストレージにおいて、両クラスタに搭載されたディスク装置４台ずつ、計８台でＲＡＩＤレベル５のＲＡＩＤを構築する。当該ＲＡＩＤデータへの更新を受けた一方のクラスタは当該データが属する第一のクラスタのキャッシュメモリに当該更新データを格納し、ホストへライト処理の完了を報告する。なお、本図ではホスト要求を受けたクラスタと当該データが属する第一のクラスタが同一である場合を想定している。当該更新データは、キャッシュメモリ上での更新データの保持数などを考慮しながら非同期にディスク装置へ書き込まれる。当該更新データがディスク装置への反映対象となった場合、対応するパリティを生成するために必要なデータをディスク装置から対応するクラスタのキャッシュメモリ１０６へそれぞれ読み上げる。簡単のため、更新されたデータストライプの更新前の値と、対応するパリティの更新前の値を読み上げるものとする。このとき、パリティがデータストライプが属する第一のクラスタでなく第二のクラスタに属する場合、第二のクラスタに対して第二のクラスタのキャッシュメモリへパリティの更新前値を読み上げる要求を送信する。当該要求を受けた第二のクラスタでは、キャッシュメモリへパリティの更新前値を読み上げる。これと並行して第一のクラスタでは当該キャッシュメモリに当該更新データストライプの更新前値が読み上げられる。必要なデータが全て揃ったら、データおよびパリティの配置から最適なパリティ生成クラスタを決定し、そのクラスタにてパリティ更新値を生成する。その後、当該更新データおよび当該更新パリティを各々が属するクラスタにより各ディスク装置へ書き込む。

【００５８】次に第２の実施形態における各処理についてフロー図を用いて詳細に説明する。

【００５９】図７はチャネルコントローラ１０３上で動作するチャネルコマンド処理の処理フロー図である。処理フローは第１の実施形態のチャネルコマンド処理のフロー図である図４と共通部分が多い。よって、相違部分であるライト処理部分について説明する。

【００６０】ステップ７０２でホスト要求がライト要求であると判定したら、ステップ７０３でステップ４０３と同様に対象クラスタのキャッシュメモリ１０６からセグメントを割当て、ライトデータを格納する。当該セグメントをダーティ状態にし、格納するデータが有効である旨、制御情報を変更したら、その時点でホストコンピュータ１００へ要求処理の完了を報告する。このとき、データ消失を避けるためには、キャッシュメモリ１０６はバッテリを用いるなどの手段により不揮発化する必要がある。

【００６１】このようにホストコンピュータ１００からの更新データはキャッシュメモリ１０６に保持され、以降、ディスクコントローラ１０４の非同期ディスク反映処理により、ホスト要求とは非同期にディスク装置１０５へ反映される。

【００６２】図８はディスクコントローラ１０４上で動作する非同期ディスク反映処理の処理フロー図である。本処理はキャッシュメモリ１０６内の更新データ量やディスクコントローラ１０４の負荷を考慮しながら起動要否を判定され、更新されてからの経過時間などを元に対象となるデータを選出され起動される。この起動要否判定、および対象データの選出は既存論理であるため、詳細説明は省略する。

【００６３】ステップ８０１で、処理対象である更新データストライプのパリティが所属するクラスタを特定する。クラスタ特定には図３で示した論理ディスクとディスク装置１０５の対応情報を用いる。本実施形態では、パリティが所属クラスタはデータの所属クラスタとは異なる場合について説明する。以後、パリティ所属クラス

タを第２のクラスタ、データ所属クラスタを第１のクラスタと呼ぶ。また、各ディスク装置を第２のディスク装置、第１のディスク装置と呼ぶ。

【００６４】ステップ８０２で、パリティ生成が必要なデータが既に各クラスタ１０２のキャッシュメモリ１０６に存在するかどうかをチェックする。本実施形態では、更新データスプライトの更新前の値と、対応するパリティの更新前の値のヒットミス判定を行う。必要なデータのヒットミス状態によりステップ８０３で分岐し、必要なデータのうちミスしているものがあれば、ステップ８０４にて不足データをキャッシュメモリ１０６に読み上げるよう、各クラスタのディスクコントローラ１０４に処理要求を送信する。例えば、更新データスプライトとパリティの更新前の値が共にミスである場合には、第１クラスタのディスクコントローラ１０４へ当該データの更新前の値を第１のディスク装置からリードする要求を、第２クラスタのディスクコントローラ１０４へ当該パリティの更新前の値を第２のディスク装置からリードする要求をそれぞれ送信する。各要求はそれぞれのディスクコマンド処理にて処理される。要求処理の完了報告を受けたら、ステップ８０５へ遷移する。

【００６５】ステップ８０５では、各クラスタのキャッシュメモリ１０６に準備したデータ/パリティ値からパリティ生成を行うクラスタ１０２を決定する。ここで決定されたパリティ生成クラスタを第３のクラスタと呼ぶ。もちろん、第３のクラスタは第１のクラスタもしくは第２のクラスタと同じである場合が多い。

【００６６】ステップ８０６では、第３のクラスタのディスクコントローラ１０４に対して、当該更新データのパリティ生成要求を送信する。当該要求の完了報告を受けたら、ステップ８０７で生成したパリティおよび当該更新データを各々第２および第１のディスク装置へ書き出す要求を第２および第１クラスタのディスクコントローラへ送信する。当該要求処理の完了報告を受けたら、本処理は終了する。

【００６７】図９は非同期ディスク反映処理の一部であるパリティ生成クラスタ決定処理の処理フロー図である。本処理では、パリティ生成に関与しているデータストライプ数やパリティ生成方式より、クラスタ間ネットワーク１０９を介したデータ通信が少なくなるようパリティ生成を行うクラスタを決定する。

【００６８】まずステップ９０１で、当該更新データストライプとパリティを共通にするデータストライプのうち、同じく更新されたものの数を算出する。複数のデータストライプが同時に更新されている場合、これらを同時にパリティ生成することで、処理コストを削減することができる。

【００６９】ステップ９０２で、１つ以上の対象更新データストライプおよびパリティについて、パリティ生成時に用いるデータ量を各クラスタ毎に算出する。例えば、図６の場合、左側の第１のクラスタに当該更新データストライプの更新値と更新前の値が、第２のクラスタには当該パリティの更新前の値に加えてパリティ生成後に更新値をキャッシュメモリ１０６に書き込む処理が発生する。よって、図６の例ではパリティ生成に際して転送が必要となるデータ量は共に２ストライプ分で等しい。だが、複数の更新データストライプを同時にパリティ生成する場合や、更新データストライプの更新値と他の全てのデータストライプの更新前値からパリティ生成する場合などには、クラスタ毎にデータ転送量の大小関係が生じる。

【００７０】ステップ９０３で、ステップ９０２で求めた値が最大となるクラスタを特定する。複数のクラスタが同じ値で最大になる可能性もある。

【００７１】ステップ９０４では、ステップ９０３で特定した１つ以上のクラスタにパリティが属するクラスタが存在するかを判定する。パリティクラスタが含まれる場合はパリティクラスタをパリティ生成を実行するクラスタ１０２に決定する(ステップ９０５)。パリティクラスタが含まれない場合は、ステップ９０３で特定した１つ以上のクラスタから任意の一つをパリティ生成クラスタに選定する。

【００７２】図１０はディスクコントローラ１０４上で動作するディスクコマンド処理の処理フロー図である。第２の実施形態では、チャネルコマンド処理および非同期ディスク反映処理より送信されたディスク装置１０５へのリード/ライト/パリティ生成要求を受信し、処理する。フロー図のリード/ライト処理の部分は図５と共通である。ただし、第１の実施形態ではＲＡＩＤレベル１を想定していたため、ホストダーティ状態のデータをミラーディスクへ書き込む処理が考えられたが、ＲＡＩＤレベル５のＲＡＩＤである第２の実施形態では、ライト対象となるのは、パリティ生成を終えた物理ダーティ状態のデータストライプとパリティである。従って、ステップ１００４では、ライトを完了したデータストライプまたはパリティのセグメント状態を物理ダーティ状態からクリーン状態に変更する。

【００７３】また、図１０ではパリティ生成要求に対する処理が追加されている。

【００７４】ステップ１００９では、パリティ生成対象である更新データセグメントおよびパリティについて、パリティ生成に必要なデータが既にキャッシュ状態に存在するかを確認する。必要なデータが既に揃っていたら、ステップ１０１０でパリティ生成を実行する。パリティ生成には、当該クラスタ１０２の実装するパリティ生成器１０８を用いる。具体的には、各データおよびパリティのセグメントからパリティ生成器の持つバッファメモリへデータを転送し、出力データをパリティ更新値を格納するためのセグメントへ転送する。

【００７５】ステップ１０１０では、パリティ生成が完

タを第２のクラスタ、データ所属クラスタを第１のクラスタと呼ぶ。また、各ディスク装置を第２のディスク装置、第１のディスク装置と呼ぶ。

【００６４】ステップ８０２で、パリティ生成が必要なデータが既に各クラスタ１０２のキャッシュメモリ１０６に存在するかどうかをチェックする。本実施形態では、更新データスプライトの更新前の値と、対応するパリティの更新前の値のヒットミス判定を行う。必要なデータのヒットミス状態によりステップ８０３で分岐し、必要なデータのうちミスしているものがあれば、ステップ８０４にて不足データをキャッシュメモリ１０６に読み上げるよう、各クラスタのディスクコントローラ１０４に処理要求を送信する。例えば、更新データスプライトとパリティの更新前の値が共にミスである場合には、第１クラスタのディスクコントローラ１０４へ当該データの更新前の値を第１のディスク装置からリードする要求を、第２クラスタのディスクコントローラ１０４へ当該パリティの更新前の値を第２のディスク装置からリードする要求をそれぞれ送信する。各要求はそれぞれのディスクコマンド処理にて処理される。要求処理の完了報告を受けたら、ステップ８０５へ遷移する。

【００６５】ステップ８０５では、各クラスタのキャッシュメモリ１０６に準備したデータ/パリティ値からパリティ生成を行うクラスタ１０２を決定する。ここで決定されたパリティ生成クラスタを第３のクラスタと呼ぶ。もちろん、第３のクラスタは第１のクラスタもしくは第２のクラスタと同じである場合が多い。

【００６６】ステップ８０６では、第３のクラスタのディスクコントローラ１０４に対して、当該更新データのパリティ生成要求を送信する。当該要求の完了報告を受けたら、ステップ８０７で生成したパリティおよび当該更新データを各々第２および第１のディスク装置へ書き出す要求を第２および第１クラスタのディスクコントローラへ送信する。当該要求処理の完了報告を受けたら、本処理は終了する。

【００６７】図９は非同期ディスク反映処理の一部であるパリティ生成クラスタ決定処理の処理フロー図である。本処理では、パリティ生成に関与しているデータストライプ数やパリティ生成方式より、クラスタ間ネットワーク１０９を介したデータ通信が少なくなるようパリティ生成を行うクラスタを決定する。

【００６８】まずステップ９０１で、当該更新データストライプとパリティを共通にするデータストライプのうち、同じく更新されたものの数を算出する。複数のデータストライプが同時に更新されている場合、これらを同時にパリティ生成することで、処理コストを削減することができる。

【００６９】ステップ９０２で、１つ以上の対象更新データストライプおよびパリティについて、パリティ生成時に用いるデータ量を各クラスタ毎に算出する。例えば、図６の場合、左側の第１のクラスタに当該更新データストライプの更新値と更新前の値が、第２のクラスタには当該パリティの更新前の値に加えてパリティ生成後に更新値をキャッシュメモリ１０６に書き込む処理が発生する。よって、図６の例ではパリティ生成に際して転送が必要となるデータ量は共に２ストライプ分で等しい。だが、複数の更新データストライプを同時にパリティ生成する場合や、更新データストライプの更新値と他の全てのデータストライプの更新前値からパリティ生成する場合などには、クラスタ毎にデータ転送量の大小関係が生じる。

【００７０】ステップ９０３で、ステップ９０２で求めた値が最大となるクラスタを特定する。複数のクラスタが同じ値で最大になる可能性もある。

【００７１】ステップ９０４では、ステップ９０３で特定した１つ以上のクラスタにパリティが属するクラスタが存在するかを判定する。パリティクラスタが含まれる場合はパリティクラスタをパリティ生成を実行するクラスタ１０２に決定する(ステップ９０５)。パリティクラスタが含まれない場合は、ステップ９０３で特定した１つ以上のクラスタから任意の一つをパリティ生成クラスタに選定する。

【００７２】図１０はディスクコントローラ１０４上で動作するディスクコマンド処理の処理フロー図である。第２の実施形態では、チャネルコマンド処理および非同期ディスク反映処理より送信されたディスク装置１０５へのリード/ライト/パリティ生成要求を受信し、処理する。フロー図のリード/ライト処理の部分は図５と共通である。ただし、第１の実施形態ではＲＡＩＤレベル１を想定していたため、ホストダーティ状態のデータをミラーディスクへ書き込む処理が考えられたが、ＲＡＩＤレベル５のＲＡＩＤである第２の実施形態では、ライト対象となるのは、パリティ生成を終えた物理ダーティ状態のデータストライプとパリティである。従って、ステップ１００４では、ライトを完了したデータストライプまたはパリティのセグメント状態を物理ダーティ状態からクリーン状態に変更する。

【００７３】また、図１０ではパリティ生成要求に対する処理が追加されている。

【００７４】ステップ１００９では、パリティ生成対象である更新データセグメントおよびパリティについて、パリティ生成に必要なデータが既にキャッシュ状態に存在するかを確認する。必要なデータが既に揃っていたら、ステップ１０１０でパリティ生成を実行する。パリティ生成には、当該クラスタ１０２の実装するパリティ生成器１０８を用いる。具体的には、各データおよびパリティのセグメントからパリティ生成器の持つバッファメモリへデータを転送し、出力データをパリティ更新値を格納するためのセグメントへ転送する。

【００７５】ステップ１０１０では、パリティ生成が完

了した後、各データおよびパリティセグメントの状態を変更する。ホストダーティ状態のデータは物理ダーティ状態に変更し、パリティセグメントも物理ダーティ状態に変更し、かつセグメント内データが有効である旨制御情報を変更する。全ての処理が完了したら、パリティ生成要求の完了を上位へ報告する(ステップ１００５)。

【００７６】なお、本発明は上記の実施形態に限定されず、その要旨の範囲内で数々の変形が可能である。 例えば、第１の実施形態と第２の実施形態で更新データのディスク装置反映タイミングを入れ替えてもよい。すなわち、第１の実施形態におけるＲＡＩＤレベル１のＲＡＩＤへの更新に対して、非同期にディスク装置への反映を行っても構わない。同様に第２の実施形態におけるＲＡＩＤレベル５のＲＡＩＤへの更新に対して、ホスト更新と同期してパリティを生成し、ディスク装置への反映を行っても構わない。

【００７７】また、データ記憶媒体をディスク装置としているが、ハードディスク装置だけでなく、ＤＶＤ－ＲＡＭやＤＶＤ－ＲＷ、ＣＤ－ＲＷ、ＭＯなどの書き込み可能なディスク媒体や、バッテリ接続などの手段により不揮発化されたメモリや、磁気テープであってもよい。

【００７８】また、本発明技術の導入を容易にするために、クラスタ内のディスク装置だけからなるＲＡＩＤ構成と、本発明のクラスタ間をまたがるＲＡＩＤ構成との間でホストコンピュータ１００からのリード/ライト要求を受けながら構成を変更することが有効である。この構成変更には、従来技術である二つの論理ディスク間でデータを入れ替える方法や、ホスト未使用の論理ディスクに対して特定論理ディスクを移動する方法を適用すればよい。論理ディスクと各クラスタの各ディスク装置との対応が第３図のような制御情報で示されていれば、既存の論理ディスクの移動方法をほぼそのまま適用することが可能である。

【００７９】

【発明の効果】本発明の計算機システムによれば、クラスタ構成ストレージを用いたクラスタ間負荷分散を実現できる。

【図面の簡単な説明】

【図１】本発明の第１の実施形態が対象とする計算機システムのブロック図である。

【図２】本発明の第１の実施形態におけるクラスタまたがりＲＡＩＤへのデータ更新処理の流れを示す概要図である。

【図３】本発明の第１の実施形態における論理ディスクとディスク装置との対応を示す制御情報例である。

【図４】本発明の第１の実施形態におけるチャネルコマンド処理のフロー図である。

【図５】本発明の第１の実施形態におけるディスクコマンド処理のフロー図である。

【図６】本発明の第２の実施形態における論理ディスクとディスク装置との対応を示す制御情報例である。

【図７】本発明の第２の実施形態におけるチャネルコマンド処理のフロー図である。

【図８】本発明の第２の実施形態における非同期ディスク反映処理のフロー図である。

【図９】本発明の第２の実施形態におけるパイティ生成クラスタ決定処理のフロー図である。

【図１０】本発明の第２の実施形態におけるディスクコマンド処理のフロー図である。

【符号の説明】

１００…ホストコンピュータ、１０１…チャネル、１０２…ストレージクラスタ、１０３…チャネルコントローラ、１０４…ディスクコントローラ、１０５…ディスク装置、１０６…キャッシュメモリ、１０７…共用メモリ、１０８…パリティ生成器、１０９…クラスタ間ネットワーク。

了した後、各データおよびパリティセグメントの状態を変更する。ホストダーティ状態のデータは物理ダーティ状態に変更し、パリティセグメントも物理ダーティ状態に変更し、かつセグメント内データが有効である旨制御情報を変更する。全ての処理が完了したら、パリティ生成要求の完了を上位へ報告する(ステップ１００５)。

【００７６】なお、本発明は上記の実施形態に限定されず、その要旨の範囲内で数々の変形が可能である。 例えば、第１の実施形態と第２の実施形態で更新データのディスク装置反映タイミングを入れ替えてもよい。すなわち、第１の実施形態におけるＲＡＩＤレベル１のＲＡＩＤへの更新に対して、非同期にディスク装置への反映を行っても構わない。同様に第２の実施形態におけるＲＡＩＤレベル５のＲＡＩＤへの更新に対して、ホスト更新と同期してパリティを生成し、ディスク装置への反映を行っても構わない。

【００７７】また、データ記憶媒体をディスク装置としているが、ハードディスク装置だけでなく、ＤＶＤ－ＲＡＭやＤＶＤ－ＲＷ、ＣＤ－ＲＷ、ＭＯなどの書き込み可能なディスク媒体や、バッテリ接続などの手段により不揮発化されたメモリや、磁気テープであってもよい。

【００７８】また、本発明技術の導入を容易にするために、クラスタ内のディスク装置だけからなるＲＡＩＤ構成と、本発明のクラスタ間をまたがるＲＡＩＤ構成との間でホストコンピュータ１００からのリード/ライト要求を受けながら構成を変更することが有効である。この構成変更には、従来技術である二つの論理ディスク間でデータを入れ替える方法や、ホスト未使用の論理ディスクに対して特定論理ディスクを移動する方法を適用すればよい。論理ディスクと各クラスタの各ディスク装置との対応が第３図のような制御情報で示されていれば、既存の論理ディスクの移動方法をほぼそのまま適用することが可能である。

【００７９】

【発明の効果】本発明の計算機システムによれば、クラスタ構成ストレージを用いたクラスタ間負荷分散を実現できる。

【図面の簡単な説明】

【図１】本発明の第１の実施形態が対象とする計算機システムのブロック図である。

【図２】本発明の第１の実施形態におけるクラスタまたがりＲＡＩＤへのデータ更新処理の流れを示す概要図である。

【図３】本発明の第１の実施形態における論理ディスクとディスク装置との対応を示す制御情報例である。

【図４】本発明の第１の実施形態におけるチャネルコマンド処理のフロー図である。

【図５】本発明の第１の実施形態におけるディスクコマンド処理のフロー図である。

【図６】本発明の第２の実施形態における論理ディスクとディスク装置との対応を示す制御情報例である。

【図７】本発明の第２の実施形態におけるチャネルコマンド処理のフロー図である。

【図８】本発明の第２の実施形態における非同期ディスク反映処理のフロー図である。

【図９】本発明の第２の実施形態におけるパイティ生成クラスタ決定処理のフロー図である。

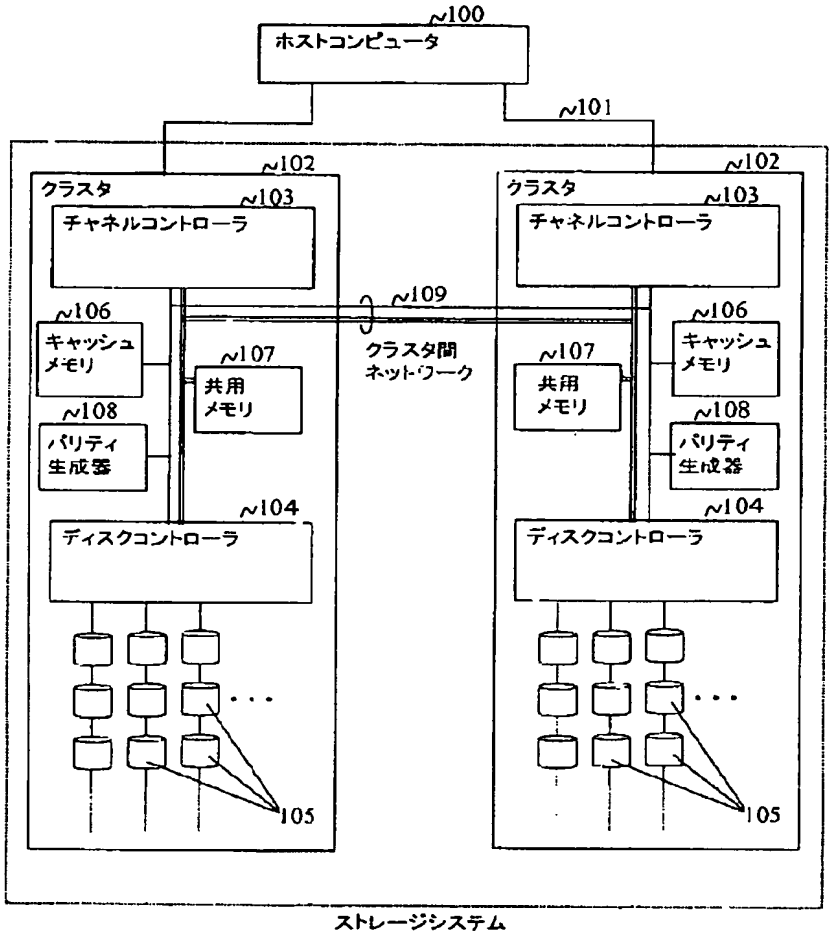
【図１０】本発明の第２の実施形態におけるディスクコマンド処理のフロー図である。

【符号の説明】

１００…ホストコンピュータ、１０１…チャネル、１０２…ストレージクラスタ、１０３…チャネルコントローラ、１０４…ディスクコントローラ、１０５…ディスク装置、１０６…キャッシュメモリ、１０７…共用メモリ、１０８…パリティ生成器、１０９…クラスタ間ネットワーク。

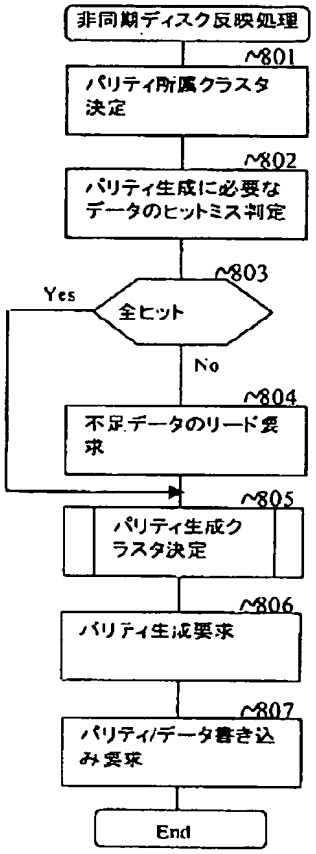
【図1】

図 1


100
ホストコンピュータ
101
102
クラスタ
103
チャネルコントローラ
106
キャッシュメモリ
107
共用メモリ
108
パリティ生成器
104
ディスクコントローラ
105
109
クラスタ間ネットワーク
ストレージシステム

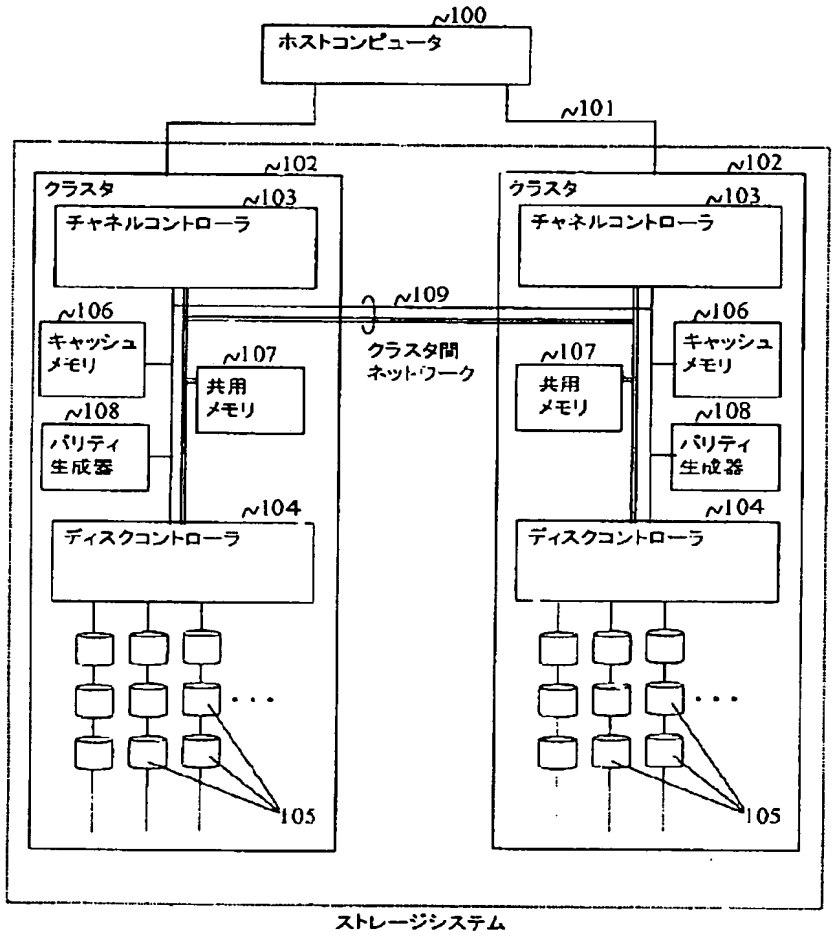
【図8】

図 8


非同期ディスク反映処理
801
パリティ所属クラスタ決定
802
パリティ生成に必要なデータのヒットミス判定
803
全ヒット
Yes
No
804
不足データのリード要求
805
パリティ生成クラスタ決定
806
パリティ生成要求
807
パリティ/データ書き込み要求
End

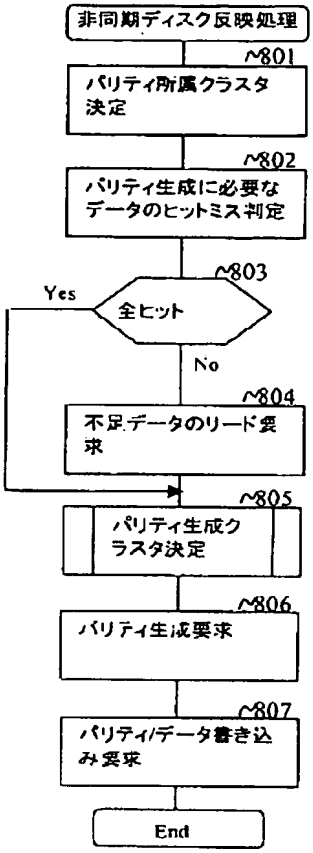
【図1】

図 1


100
ホストコンピュータ
101
102
クラスタ
103
チャネルコントローラ
106
キャッシュメモリ
107
共用メモリ
108
パリティ生成器
104
ディスクコントローラ
105
109
クラスタ間ネットワーク
ストレージシステム

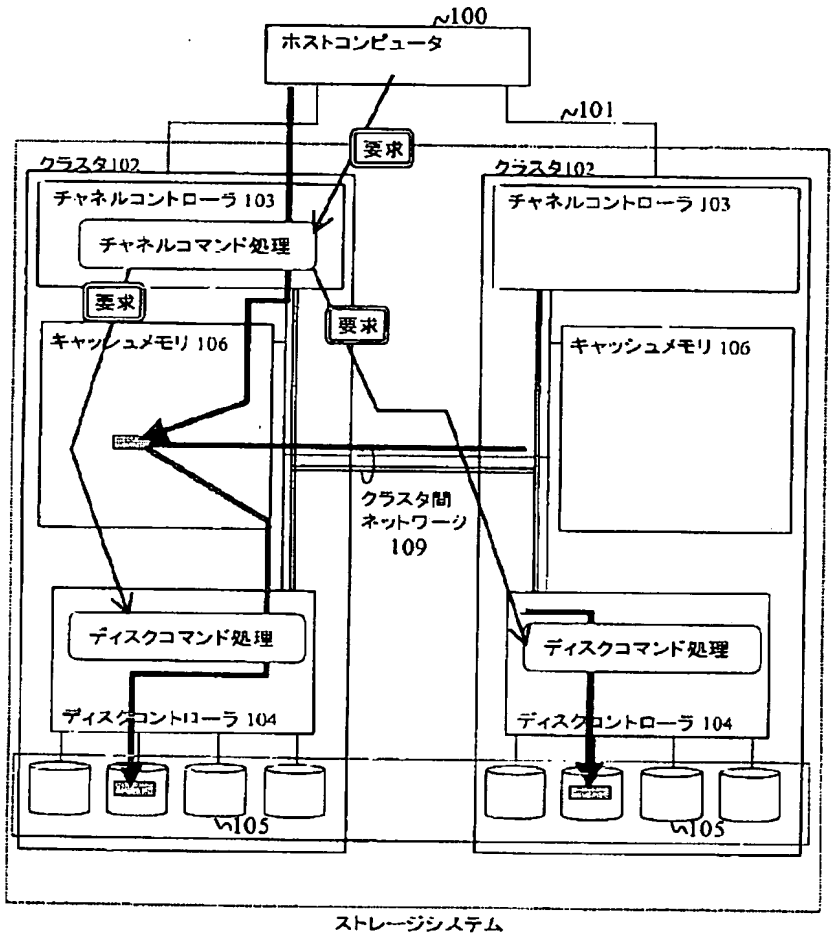
【図8】

図 8


非同期ディスク反映処理
801
パリティ所属クラスタ決定
802
パリティ生成に必要なデータのヒットミス判定
803
Yes
全ヒット
No
804
不足データのリード要求
805
パリティ生成クラスタ決定
806
パリティ生成要求
807
パリティ/データ書き込み要求
End

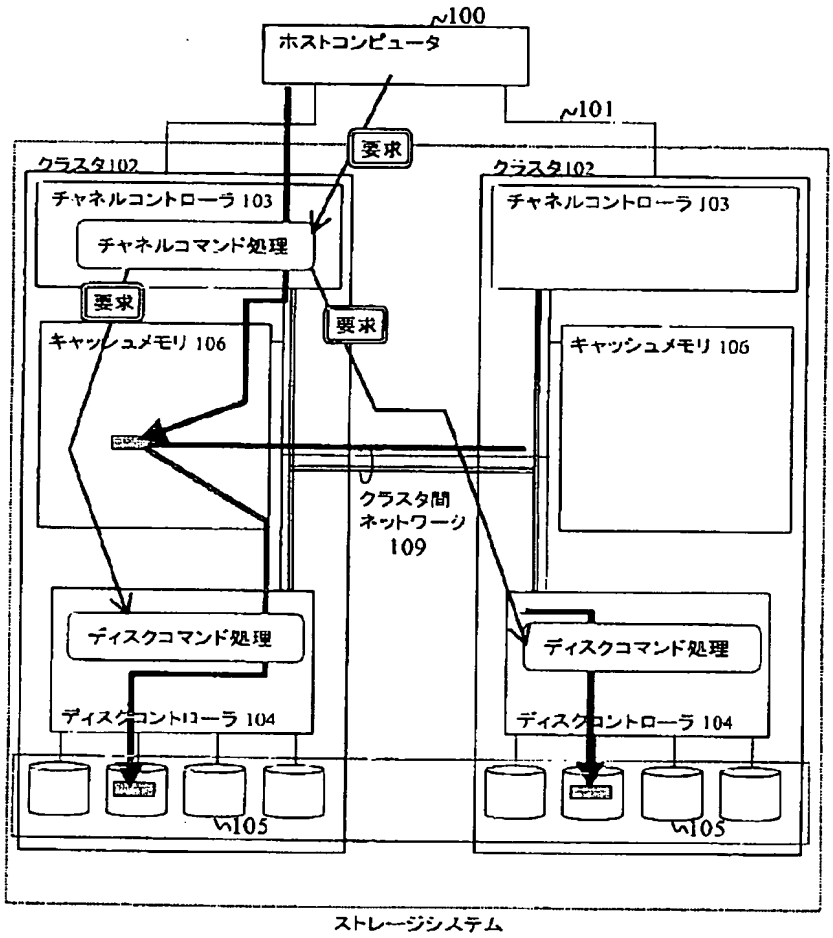
【図2】

図 2


100
ホストコンピュータ
101
要求
クラスタ102
チャネルコントローラ 103
チャネルコマンド処理
要求
キャッシュメモリ 106
要求
クラスタ102
チャネルコントローラ 103
キャッシュメモリ 106
クラスタ間
ネットワーク
109
ディスクコマンド処理
ディスクコントローラ 104
105
ディスクコマンド処理
ディスクコントローラ 104
105
ストレージシステム

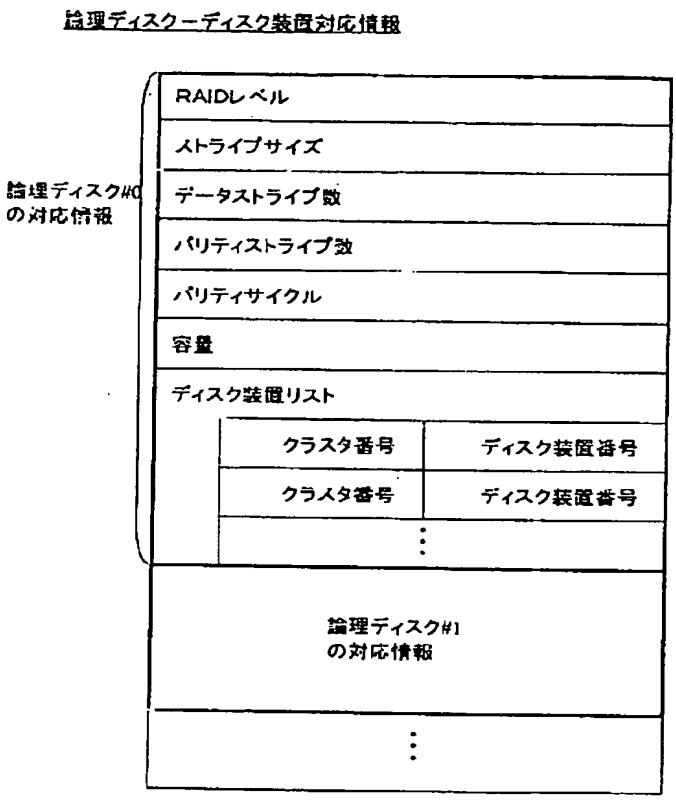
【図2】

図 2


100
ホストコンピュータ
101
要求
クラスタ102
チャネルコントローラ 103
チャネルコマンド処理
要求
キャッシュメモリ 106
要求
クラスタ102
チャネルコントローラ 103
キャッシュメモリ 106
クラスタ間
ネットワーク
109
ディスクコマンド処理
ディスクコントローラ 104
ディスクコマンド処理
ディスクコントローラ 104
105
105
ストレージシステム

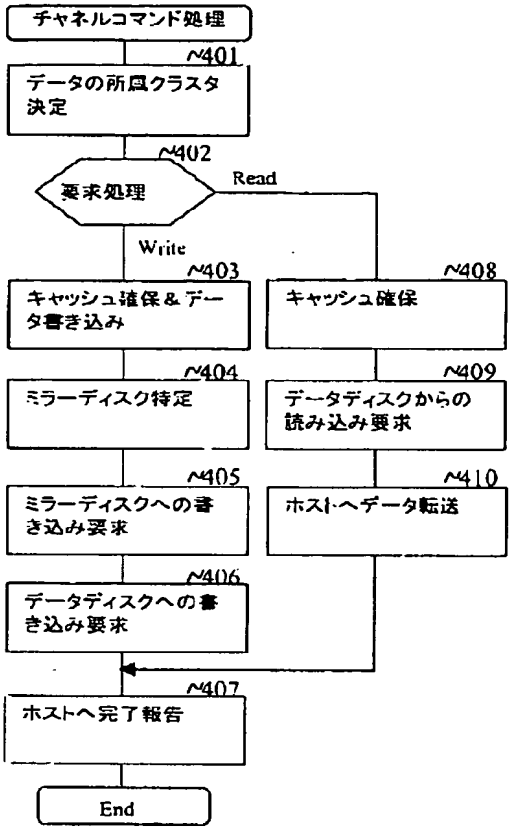
【図3】

図 3


論理ディスク−ディスク装置対応情報
論理ディスク#0の対応情報
RAIDレベル
ストライプサイズ
データストライプ数
パリティストライプ数
パリティサイクル
容量
ディスク装置リスト
クラスタ番号
ディスク装置番号
クラスタ番号
ディスク装置番号
論理ディスク#1の対応情報

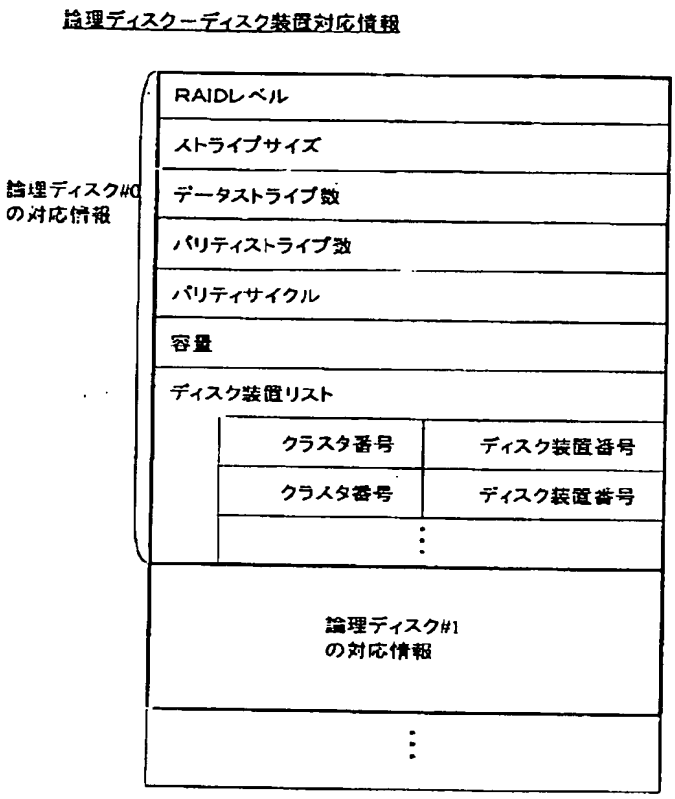
【図4】

図 4


チャネルコマンド処理
401
データの所属クラスタ決定
402
要求処理
Read
Write
403
キャッシュ確保＆データ書き込み
404
ミラーディスク特定
405
ミラーディスクへの書き込み要求
406
データディスクへの書き込み要求
408
キャッシュ確保
409
データディスクからの読み込み要求
410
ホストへデータ転送
407
ホストへ完了報告
End

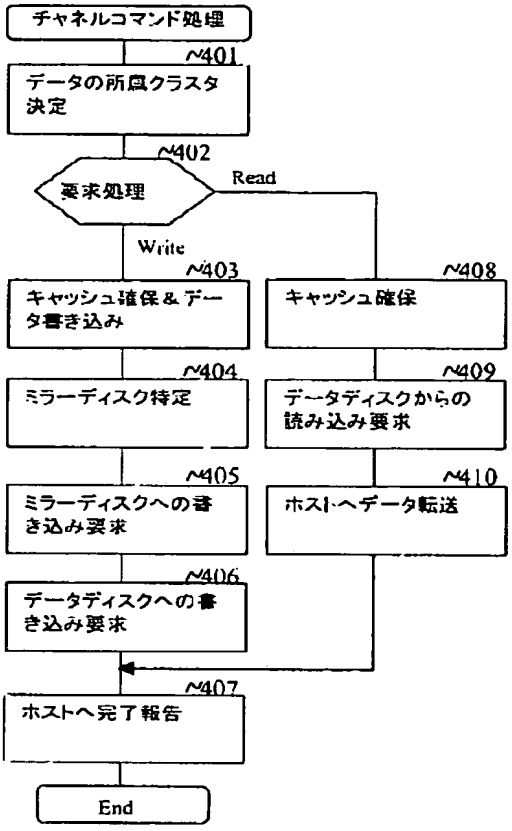
【図3】

図 3


論理ディスク－ディスク装置対応情報
論理ディスク#0の対応情報
RAIDレベル
ストライプサイズ
データストライプ数
パリティストライプ数
パリティサイクル
容量
ディスク装置リスト
クラスタ番号
ディスク装置番号
クラスタ番号
ディスク装置番号
論理ディスク#1の対応情報

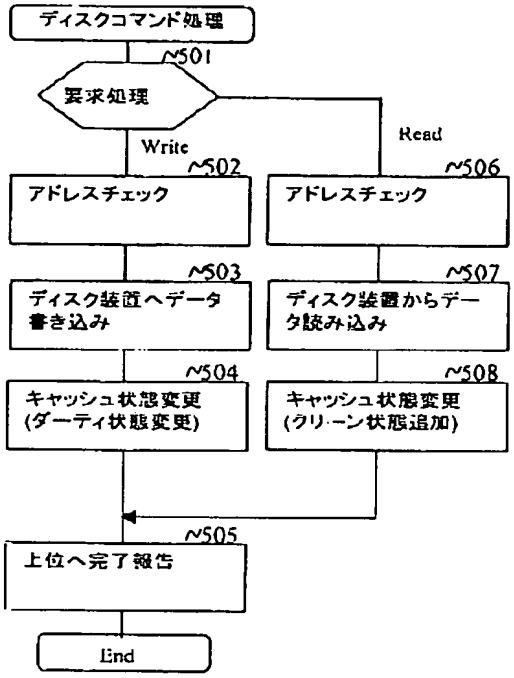
【図4】

図 4


チャネルコマンド処理
401
データの所属クラスタ決定
402
要求処理
Read
Write
403
キャッシュ確保＆データ書き込み
404
ミラーディスク特定
405
ミラーディスクへの書き込み要求
406
データディスクへの書き込み要求
408
キャッシュ確保
409
データディスクからの読み込み要求
410
ホストへデータ転送
407
ホストへ完了報告
End

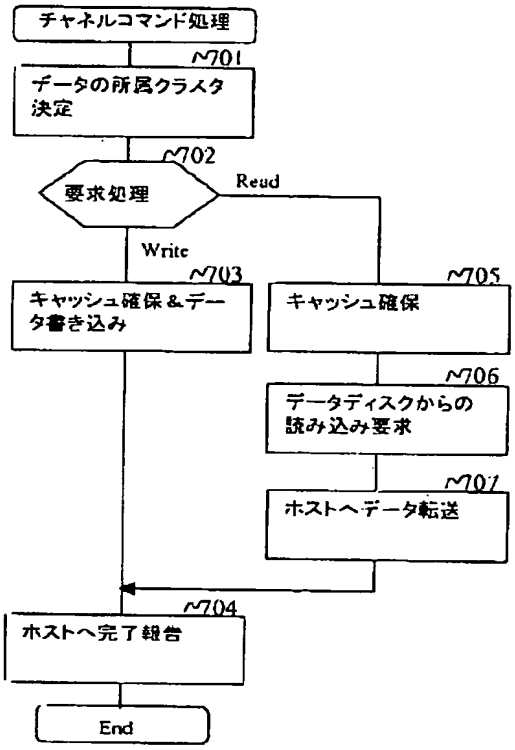
【図5】

図 5


ディスクコマンド処理
501
要求処理
Write
Read
502
アドレスチェック
506
アドレスチェック
503
ディスク装置へデータ書き込み
507
ディスク装置からデータ読み込み
504
キャッシュ状態変更(ダーティ状態変更)
508
キャッシュ状態変更(クリーン状態追加)
505
上位へ完了報告
End

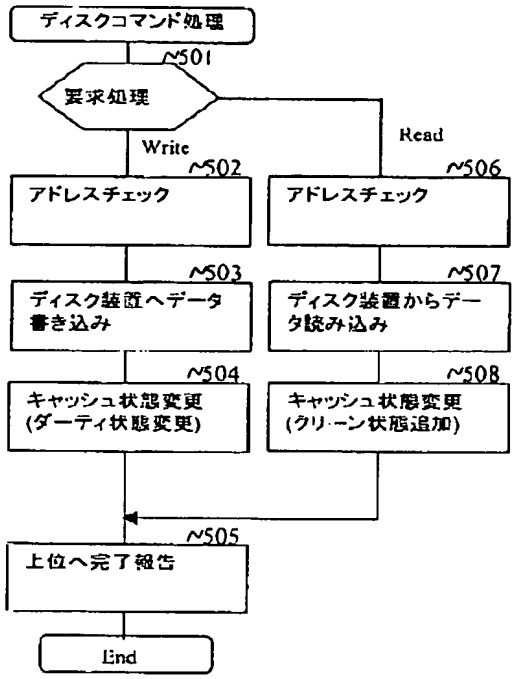
【図7】

図 7


チャネルコマンド処理
701
データの所属クラスタ決定
702
要求処理
Read
Write
703
キャッシュ確保&データ書き込み
705
キャッシュ確保
706
データディスクからの読み込み要求
707
ホストへデータ転送
704
ホストへ完了報告
End

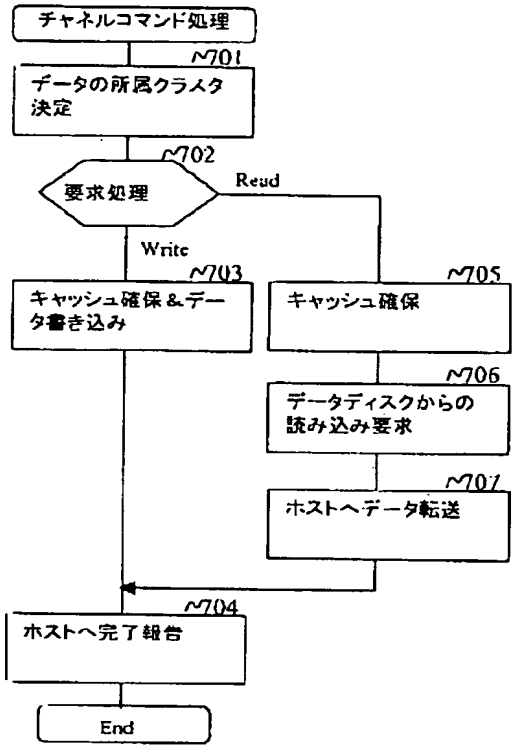
【図5】

図 5


ディスクコマンド処理
501
要求処理
Write
Read
502
アドレスチェック
506
アドレスチェック
503
ディスク装置へデータ書き込み
507
ディスク装置からデータ読み込み
504
キャッシュ状態変更(ダーティ状態変更)
508
キャッシュ状態変更(クリーン状態追加)
505
上位へ完了報告
End

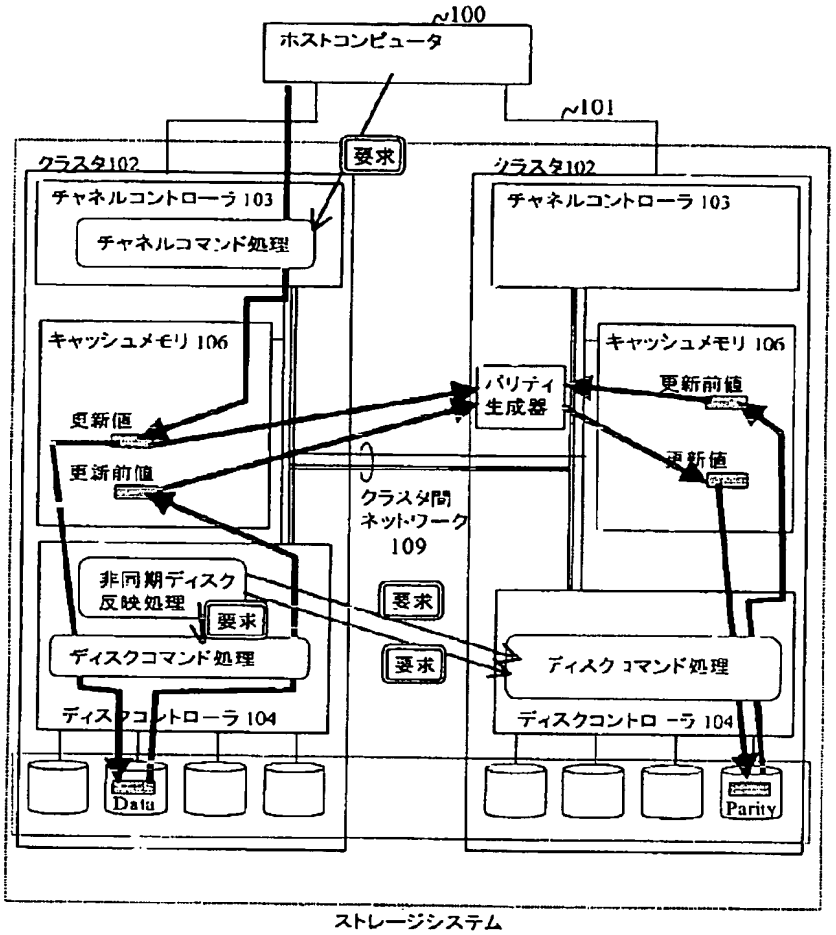
【図7】

図 7


チャネルコマンド処理
701
データの所属クラスタ決定
702
要求処理
Read
Write
703
キャッシュ確保&データ書き込み
705
キャッシュ確保
706
データディスクからの読み込み要求
707
ホストへデータ転送
704
ホストへ完了報告
End

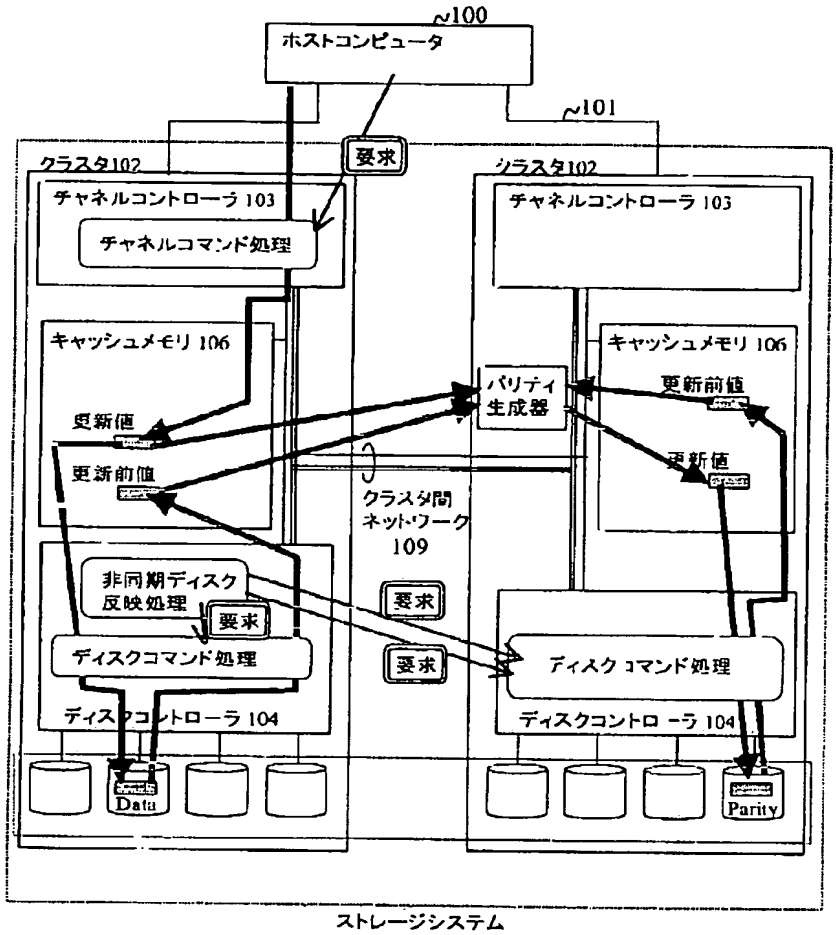
【図6】

図 6


~100
ホストコンピュータ
~101
要求
クラスタ102
チャネルコントローラ 103
チャネルコマンド処理
キャッシュメモリ 106
更新値
更新前値
パリティ生成器
クラスタ間ネットワーク 109
非同期ディスク反映処理
要求
ディスクコマンド処理
ディスクコントローラ 104
Data
Parity
ストレージシステム

【図6】

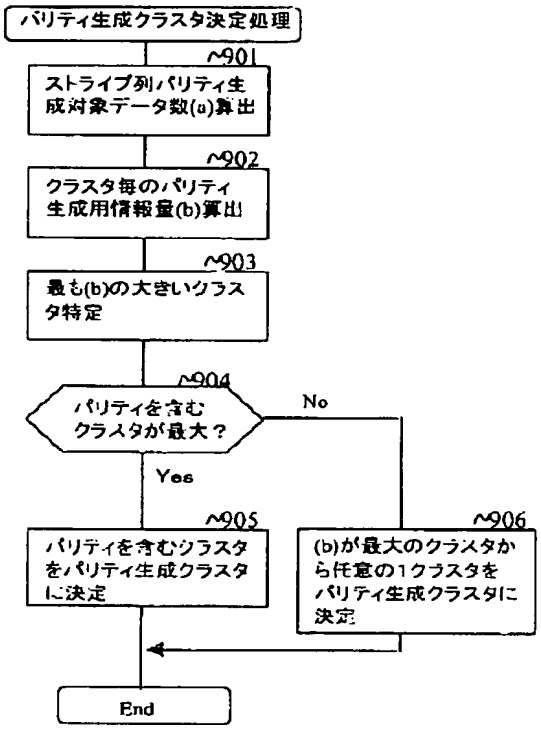
図 6



ストレージシステム

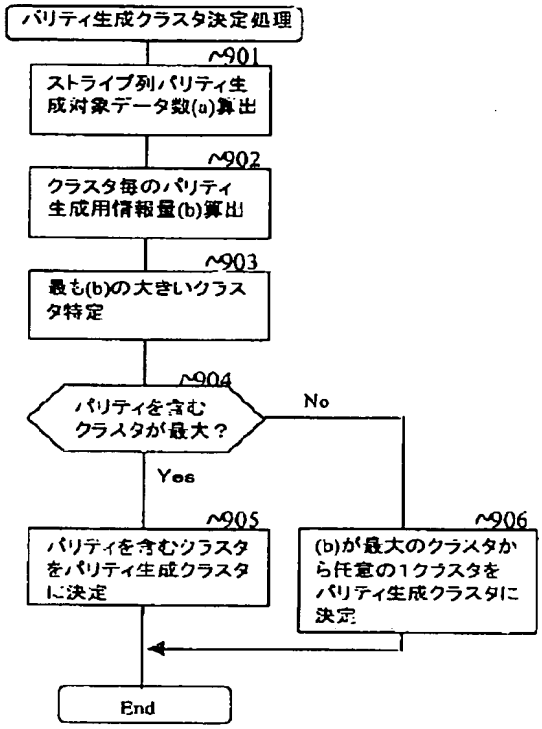
【図9】

図 9


パリティ生成クラスタ決定処理
~901
ストライプ列パリティ生成対象データ数(a)算出
~902
クラスタ毎のパリティ生成用情報量(b)算出
~903
最も(b)の大きいクラスタ特定
~904
パリティを含むクラスタが最大？
No
Yes
~905
パリティを含むクラスタをパリティ生成クラスタに決定
~906
(b)が最大のクラスタから任意の1クラスタをパリティ生成クラスタに決定
End

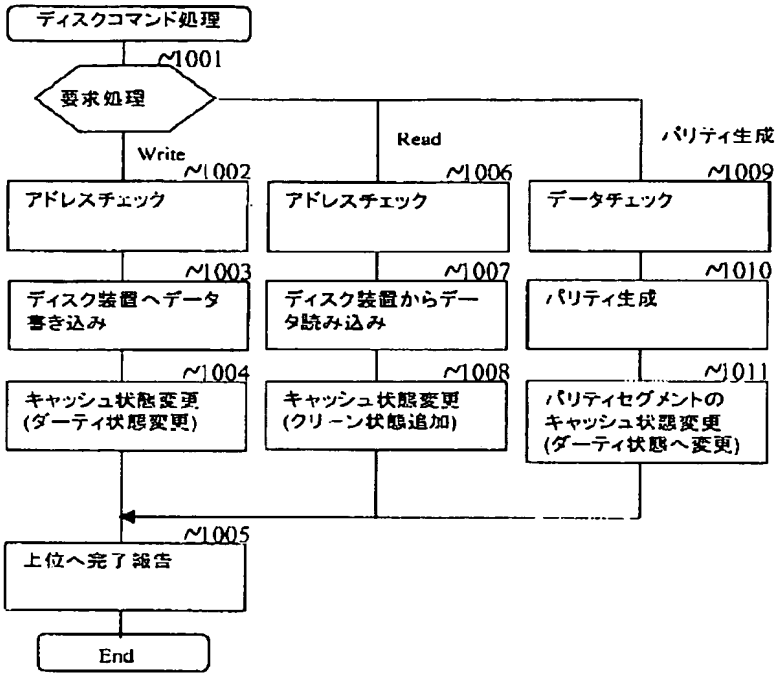
【図９】

図 9


パリティ生成クラスタ決定処理
~901
ストライプ列パリティ生成対象データ数(a)算出
~902
クラスタ毎のパリティ生成用情報量(b)算出
~903
最も(b)の大きいクラスタ特定
~904
パリティを含むクラスタが最大？
No
Yes
~905
パリティを含むクラスタをパリティ生成クラスタに決定
~906
(b)が最大のクラスタから任意の1クラスタをパリティ生成クラスタに決定
End

【図１０】

図 10



フロントページの続き

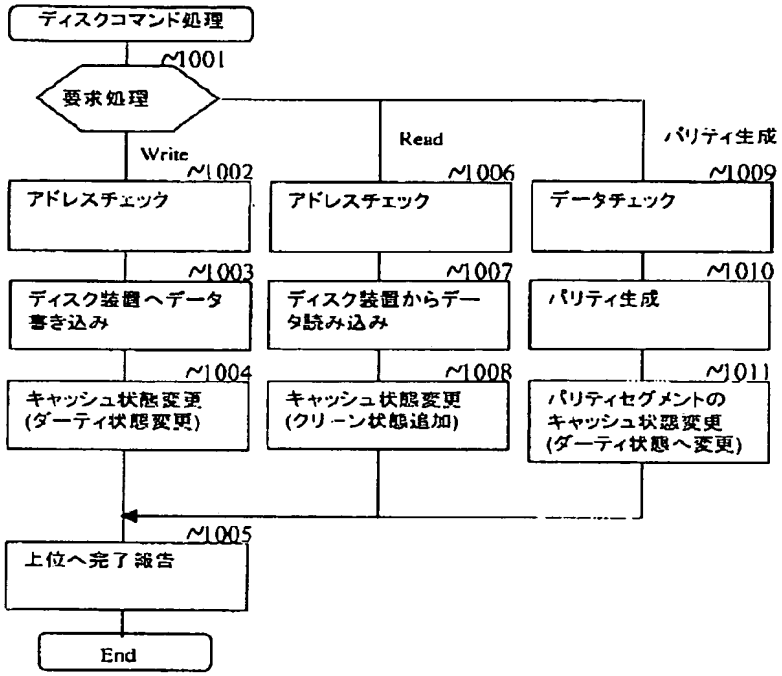
(51)Int.Cl.[7]	識別記号	FI	(参考)
G06F 12/08	557	G06F 12/08	557

(72)発明者 佐藤 孝夫
神奈川県小田原市中里322番地２号 株式会社日立製作所ＲＡＩＤシステム事業部内

Ｆターム(参考) 5B005 JJ11 KK13 MM11
5B065 BA01 CA12 CA30 CC03 CE12 CH01 CS01 EA03 EA12

【図１０】

図 10



フロントページの続き

(51)Int.Cl.[7]	識別記号	FI	(参考)
G06F 12/08	557	G06F 12/08	557

(72)発明者 佐藤 孝夫
神奈川県小田原市中里322番地２号 株式会社日立製作所ＲＡＩＤシステム事業部内

Ｆターム(参考) 5B005 JJ11 KK13 MM11
5B065 BA01 CA12 CA30 CC03 CE12 CH01 CS01 EA03 EA12